VIVEKANANDHA

COLLEGE OF ARTS AND SCIENCES FOR WOMEN

ELAYAMPALAYAM, TIRUCHENGODE (Tk.), NAMAKKAL (Dt.).

(Affiliated to Periyar University, Approved by AICTE & Re-Accredited with A Grade by NAAC)

PG AND RESEARCH

DEPARTMENT OF COMPUTER SCIENCE & APPLICATIONS

B.Sc. COMPUTER SCIENCE

SYLLABUS & REGULATIONS

FOR CANDIDATES ADMITTED FROM 2023-24 ONWARDS

UNDER AUTONOMOUS & OBE PATTERN

VIVEKANANDHA EDUCATIONAL INSTITUTIONS

Angammal Educational Trust

Elayampalayam, Tiruchengode (Tk.), Namakkal (DT)

VIVEKANANDHA COLLEGE OF ARTS AND SCIENCES FOR WOMEN

(AUTONOMOUS)

B.Sc (COMPUTER SCIENCE)

(Candidates admitted from 2023-2024 onwards)

REGULATIONS

I. SCOPE OF THE PROGRAMME

Bachelor of Computer Science can be considered to be one of the most prominent UG level programs in our country. This program mainly deals with the development of computer applications for the purpose of updating computer programming languages. B.Sc.[CS] also aims at creating strong knowledge of theoretical Information Technology subjects who can be employed in software development and testing units of industries. The course has a time period of 3 years with 6 semesters.

II. SALIENT FEATURES

- Regular conduct of guest lectures and seminars
- Campus recruitment
- Provides facilities such as Internet Access and In-House Library
- Provides Career Guidance for Post Graduate Courses like M.Sc, MCA and the Certifications in programming languages
- Conduct of Personality Development Program
- Visiting Faculties from Industries

III. OBJECTIVES OF THE PROGRAMME

The Course Objective of the B.Sc. Computer Science program is to provide advanced and in-depth knowledge of Information Technology and its applications to enable students pursue a professional career in Information and Communication Technology in related industry, business and research. The course designed to impact professional knowledge and practical skills to the students.

IV. ELIGIBILITY FOR ADMISSION

A Candidates seeking admission to the first year Degree course (B.Sc. Computer Science)shall be required to have passed Higher Secondary Examination with Mathematics or Business

Mathematics or Computer Science or Computer Applications or Computer Technology or Statistics (Academic Stream or Vocational Stream) as one of the subject under Higher Secondary Board of Examination, conducted by the Government of Tamilnadu or an examination accepted as equivalent there to by the syndicate, subject to such conditions as may be prescribed there to are permitted to appear and qualify for the B.Sc. Computer Science Degree Examination of Periyar University after a course of study of three academic years.

V. DURATION OF THE PROGRAMME

- The course shall extend over a period of three academic years consisting of six semesters. Each academic year will be divided into two semesters. The First semester will consist of the period from July to November and the Second semester from December to April.
- The subjects of the study shall be in accordance with the syllabus prescribed from time to time by the Board of Studies of Vivekanandha College of Arts and Sciences for Women with the approval of Periyar University.

VI. CONTINUOUS INTERNAL ASSESSMENT (CIA)

The performance of the students will be assessed continuously and the Internal

ASSESSMENT MARKS **FOR THEORY PAPERS** WILL BE AS UNDER:

1	Average of Two Tests	-	05
2	Model Exam	-	10
3	Assignment	-	05
4	Attendance	-	05
	To	-	25

ASSESSMENT MARKS **FOR PRACTICAL PAPERS** WILL BE AS UNDER:

1	Model Exam	-	20
2	Observation Note	-	10

3	Attendance	-	10
	To	-	40

PASSING MINIMUM - EXTERNAL

THEORY	In the End Semester Examinations, the passing minimum shall be 40% Out of 75 Marks. (30 Marks)
PRACTICAL / MINI PROJECT	In the End Semester Examinations, the passing minimum shall be 40% Out of 60 Marks. (24 Marks)

VII. ELIGIBILITY FOR EXAMINATION

A candidate will be permitted to appear for the University Examination only on learning 75 % of attendance and only when her conduct has been satisfactory. It shall be open to grant exemption to a candidate for valid reasons subject to conditions prescribed.

DISTRIBUTION OF MARKS FOR ATTENDANCE:

ATTENDANCE PERCENTAGE	MARKS	
	THEORY	PRACTICAL
75-80	1	2
81-85	2	4
86-90	3	6
91-95	4	8
96-100	5	10

VIII. CLASSIFICATION OF SUCCESSFUL CANDIDATES

Successful candidates passing the Examination of Core Courses (Main & Allied Subjects) &Securing Marks.

a) 75 % and above shall be declared to have passed the examination in First Class with Distinction provided they pass all the examinations prescribed for the course at first appearance itself.
b) 60% and above but below 75 % shall be declared to have passed the Examinations in First Class.
c) 50% & above but below 60% shall be declared to have passed the examinations in Second Class.
d) All the remaining successful candidates shall be declared to have passed the examinations in Third Class.
e) Candidates who pass all the examinations prescribed for the course at the First appearance itself and within a period of three Consecutive Academic years from the year of admission only will be eligible for University Rank.

IX. ELIGIBILITY FOR AWARD OF THE DEGREE

A candidate shall be eligible for the award of the Degree only if she has undergone theabove Degree for a period of not less than Three Academic years comprising of six semesters and passed the Examinations prescribed and fulfilled such conditions has have been prescribed therefore.

X. PROCEDURE IN THE EVENT OF FAILURE

If a candidate fails in a particular subject, she may reappear for the university examinationin the concerned subject in subsequent semesters and shall pass the examination.

XI. COMMENCEMENT OF THESE REGULATIONS

These regulations shall take effect from the academic year 2023-2024 (i.e.,) for the students who are to be admitted to the First year of the course during the Academic year 2023-24 and thereafter.

XII. TRANSITORY PROVISIONS

Candidates who were admitted to the UG course of study before 2021-2022 shall be permitted to appear for the examinations under those regulations for the period of Three years ie. Up to and inclusive of the Examinations of 2023-2024. Thereafter, they will be permitted to appear for the examinations only under the regulations then in force.

EVALUATION OF EXTERNAL EXAMINATIONS (EE)

QUESTION PAPER PATTERN – Theory	
Time duration: 3 Hours	**Max. Marks: 75**
PART- A: (10 x 1= 10)	Answer all the Questions Two Questions from each Unit
PART- B: (5x 7 = 35)	Answer all the questions One Question from each Unit (Either or Type)
PART- C: (3x 10 = 30)	Answer any **THREE** of the questions One Question from each Unit (3 Out of 5)
IN THE END SEMESTER EXAMINATIONS, THE PASSING MINIMUM SHALL BE **40% OUT OF 75 MARKS. (30 MARKS)**	

QUESTION PAPER PATTERN – Practical	
Time duration: 3 Hours	**Max. Marks: 60**
1. One compulsory question from the given list of objectives	**30 Marks**
2. One either/or type question from the given list of objectives	**30 Marks**
IN THE END SEMESTER EXAMINATIONS, THE PASSING MINIMUM SHALL BE40% OUT OF 60 MARKS. (24 MARKS)	

First Year (Semester – I)

Course Code: 23U1CSC01	Python Programming		Credits 04
Lecture Hours per week - 4	**Tutorial Hours :75 (T) per week**	**Lab Practice Hours: (P)per week**	**Total: (L+T+P) per week: 4**
Course Category : Core	**Year & Semester: I Year I Semester**	**Admission Year: 2023-2024**	
Pre-requisite	Basic Knowledge of Programming concept		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

- Describe the core syntax and semantics of Python programming language.
- Discover the need for working with the strings and functions.
- Illustrate the process of structuring the data using lists, dictionaries, tuples and sets.
- Understand the usage of packages and Dictionaries

Course Outcomes: (for students: To know what they are going to learn)

CO1: Develop and execute simple Python programs

CO2: Write simple Python programs using conditionals and looping for solving problems

CO3: Decompose a Python program into functions

CO4: Represent compound data using Python lists, tuples, dictionaries etc.

CO5: Read and write data from/to files in Python programs

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	Contents	RequiredHours
I	Introduction: The essence of computational problem solving – Limits of computational problem solving-Computer algorithms-Computer Hardware-Computer Software-The process of computational problem solving-Python programming language - Literals - Variables and Identifiers - Operators - Expressions and Data types, Input / output	**12**
II	Control Structures: Boolean Expressions - Selection Control - If Statement- Indentation in Python- Multi-Way Selection -- Iterative Control- While Statement- Infinite loops- Definite vs. Indefinite Loops- Boolean Flag. String, List and Dictionary, Manipulations Building blocks of python programs, Understanding and using ranges.	**12**
III	Functions: Program Routines- Defining Functions- More on Functions	**12**

	:Calling Value-Returning Functions- Calling Non-Value-Returning Functions- Parameter Passing - Keyword Arguments in Python - Default Arguments in Python-Variable Scope. Recursion: Recursive Functions	
IV	Objects and their use: Software Objects - Turtle Graphics – Turtle attributes-Modular Design: Modules - Top-Down Design - Python Modules - Text Files: Opening, reading and writing text files – Database Programming: Connecting to a database, Creating Tables, INSERT, UPDATE, DELETE and READ operations, Transaction Control, Disconnecting from a database, String Processing - Exception Handling	**12**
V	Dictionaries and Sets: Dictionary type in Python - Set Data type. Object Oriented Programming using Python: Encapsulation - Inheritance – Polymorphism. Python packages: Simple programs using the built-in functions of packages matplotlib, numpy, pandas etc.	**12**

Learning Resources:

- **Recommended Texts**

1. Charles Dierbach, "Introduction to Computer Science using Python - A computational Problem solving Focus", Wiley India Edition, 2015.

2. Wesley J. Chun, "Core Python Applications Programming", 3rd Edition , Pearson Education, 2016

- **Reference Books**

1. Mark Lutz, "Learning Python Powerful Object Oriented Programming", O'reilly Media 2018, 5th Edition.
2. Timothy A. Budd, "Exploring Python", Tata MCGraw Hill Education Private Limited 2011, 1 st Edition.
3. John Zelle, "Python Programming: An Introduction to Computer Science", Second edition, Course Technology Cengage Learning Publications, 2013, ISBN 978-1590282410

- Michel Dawson, "Python Programming for Absolute Beginers" , Third Edition, Course Technology Cengage Learning Publications, 2013, ISBN 978-1435455009

- **Web resources**

https://onlinecourses.swayam2.ac.in/cec22_cs20/preview

Mapping with Programme Outcomes:

Map course outcomes for each course with programme outcomes (PO) in the 3-point scale of Strong, Medium and Low

CO/PSO	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5	PSO 6
CO 1	3	3	3	3	3	3
CO 2	3	3	3	3	3	3
CO 3	2	2	3	2	2	3
CO 4	3	3	3	3	3	2
CO 5	3	3	3	3	2	3
Weightage of course contributedto each PSO	14	14	15	14	13	14

Course Code: 23U1CSCP01	Python Programming Lab		Credits : 4
Lecture Hours: (L) per week: 05	**Tutorial Hours : (T) per week**	**Lab Practice Hours: 5 per week**	**Total: (L+T+P) per week: 5**
Course Category :Core	**Year & Semester: I Year I Semester**		**Admission Year: 2023-2024**
Pre-requisite	Basic of programming skill		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

- Acquire programming skills in core Python.
- Acquire Object-oriented programming skills in Python.
- Develop the skill of designing graphical-user interfaces (GUI) in Python.
- Develop the ability to write database applications in Python.
- Acquire Python programming skills to move into specific branches

Course Outcomes: (for students: To know what they are going to learn)

CO1: To understand the problem solving approaches

CO2: To learn the basic programming constructs in Python

CO3: To practice various computing strategies for Python-based solutions to real world problems

CO4: To use Python data structures - lists, tuples, dictionaries.

CO5: To do input/output with files in Python.

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

	List of Exercises:	**Required Hours**
	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 five subjects are to be input by 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, to find the area of rectangle, square, circle and triangle by accepting suitable input parameters from user. 4. Write a Python script that prints prime numbers less	

	than 20. 5. Program to find factorial of the given number using recursive function. 6. Write a Python program to count the number of even and odd numbers from array of N numbers. 7. Write a Python class to reverse a string word by word. 8. Given a tuple and a list as input, write a program to count the occurrences of all items of the list in the tuple. (Input : tuple = ('a', 'a', 'c', 'b', 'd'), list = ['a', 'b'], Output : 3) 9. Create a Savings Account class that behaves just like a BankAccount, but also has an interest rate and a method that increases the balance by the appropriate amount of interest (Hint:use Inheritance). 10. Write a Python program to construct the following pattern, using a nested loop * ** *** **** ***** **** *** ** * 11. Read a file content and copy only the contents at odd lines into a new file. 12. Create a Turtle graphics window with specific size. 13. Write a Python program for Towers of Hanoi using recursion 14. Create a menu driven Python program with a dictionary for words and their meanings.	

	15. Devise a Python program to implement the Hangman Game.	

Learning Resources:Recommended Texts

1. Charles Dierbach, "Introduction to Computer Science using Python - A computational Problem solving Focus", Wiley India Edition, 2015.

2. Wesley J. Chun, "Core Python Applications Programming", 3rd Edition , Pearson Education, 2016

- **Reference Books**

1. Mark Lutz, "Learning Python Powerful Object Oriented Programming", O'reilly Media 2018, 5th Edition.

2. Timothy A. Budd, "Exploring Python", Tata MCGraw Hill Education Private Limited 2011, 1 st Edition.

3. John Zelle, "Python Programming: An Introduction to Computer Science", Second edition, Course Technology Cengage Learning Publications, 2013, ISBN 978- 1590282410

4. 4.Michel Dawson, "Python Programming for Absolute Beginers" , Third Edition, Course Technology Cengage Learning Publications, 2013, ISBN 978-435455009

Course Code: FC1 23U1CSS01	Problem Solving Techniques		Credits: 2
Lecture Hours: (L) per week: 2	Tutorial Hours : (T) per week	Lab Practice Hours: (P)per week	Total: (L+T+P) per week: 2
Course Category : FC	Year & Semester: I Year I Semester	Admission Year: 2023-24	
Pre-requisite	Basic of Problem-solving skills		

Learning Objectives:

- To understand the importance of algorithms and programs, and to know of the basic problem solving strategies.
- To learn efficient strategies and algorithms to solve standard problems, thus laying a firm foundation for designing algorithmic solutions to problems.

Course Outcomes: (for students: To know what they are going to learn)

CO1: Understand the systematic approach to problem solving.

CO2: Know the approach and algorithms to solve specific fundamental problems.

CO3: Understand the efficient approach to solve specific factoring-related problems.

CO4: Understand the efficient array-related techniques to solve specific problems.

CO5: Understand the efficient methods to solve specific problems related to text processing.

Understand how recursion works.

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	Contents	Required Hours
I	**Introduction:** Notion of algorithms and programs – Requirements for solving problems by computer – The problem-solving aspect: Problem definition phase, Getting started on a problem, The use of specific examples, Similarities among problems, Working backwards from the solution – General problem-solving strategies - Problem solving using top-down design – Implementation of algorithms – The concept of Recursion.	06
II	**Fundamental Algorithms**: Exchanging the values of two variables – Counting - Summation of a set of numbers - Factorial computation - Sine function computation - Fibonacci Series generation - Reversing the digits of an integer – Base Conversion.	06
III	**Factoring Methods**: Finding the square root of a number – The smallest divisor of an integer – Greatest common divisor of two integers - Generating prime numbers – Computing the prime factors of an integer – Generation of pseudo-random numbers - Raising a number to a large power – Computing the *n*th Fibonacci number.	06
IV	**Array Techniques**: Array order reversal – Array counting or histogramming – Finding the maximum number in a set - Removal	06

	of duplicates from an ordered array - Partitioning an array – Finding the k^{th} smallest element – Longest monotone subsequence.	
V	**Text Processing and Pattern Searching**: Text line length adjustment – Left and right justification of text – Keyword searching in text – Text line editing – Linear pattern search. **Recursive algorithms**: Towers of Hanoi – Permutation generation.	**06**

Learning Resources:

- **Recommended Texts**

1. R. G. Dromey, *How to Solve it by Computer*, Pearson India, 2007.

- **Reference Books**

1. George Polya, Jeremy Kilpatrick, *The Stanford Mathematics Problem Book: With Hints and Solutions*, Dover Publications, 2009 (Kindle Edition 2013).
2. Greg W. Scragg, *Problem Solving with Computers*, Jones & Bartlett 1st edition, 1996.

Mapping with Programme Outcomes:

Map course outcomes for each course with programme outcomes (PO) in the 3-point scale of Strong, Medium and Low

CO/PSO	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5	PSO 6
CO 1	3	3	3	3	3	3
CO 2	3	3	3	2	3	3
CO 3	3	3	2	3	2	2
CO 4	3	3	3	3	3	3
CO 5	2	3	2	3	2	3
Weightage of course contributedto each PSO	14	15	13	14	13	14

First Year (Semester – II)

Course Code: CC3 23U2CSC02	Data Structures & Algorithms		Credits: 4
Lecture Hours: (L) per week: 4	Tutorial Hours : (T) per week	Lab Practice Hours: (P)per week	Total: (L+T+P) per week: 4
Course Category : CC3	Year & Semester: I Year II Semester		Admission Year: 2023-2024
Pre-requisite	Basic knowledge in data and representations		
Links to other Courses			

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

- To impart the basic concepts of data structures and algorithms.
- To acquaint the student with the basics of the various data structures and make the students knowledgeable in the area of data structures.
- This course also gives insight into the various algorithm design techniques

Course Outcomes: (for students: To know what they are going to learn)

CO1: To introduce the concepts of Data structures and to understand simple linear data structures.

CO2: Learn the basics of stack data structure, its implementation and application

CO3: Use the appropriate data structure in context of solution of given problem and demonstrate a familiarity with major data structures.

CO4: To introduce the basic concepts of algorithms

CO5: To give clear idea on algorithmic design paradigms like Dynamic Programming, Backtracking, Branch and Bound

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	Contents	Required Hours
I	**INTRODUCTION TO DATA STRUCTURES:** • Data Structures: Definition- Time & Space Complexity, • Arrays: Representation of arrays, Applications of arrays, sparse matrix and its representation, • Linear list: Singly linked list implementation, insertion, deletion and searching operations on linear list • Circular linked list: implementation, Double linked list implementation, insertion, deletion and searching operations. Applications of linked lists- Dynamic Storage management.	12

II	**STACKS:** • Operations, array and linked representations of stack, • stack applications, infix to postfix conversion, postfix expression evaluation, recursion implementation	**12**
III	**QUEUES, TREES & GRAPHS:** • **Queues**: operations on queues, array and linked representations. • **Circular Queue:** operations,, applications of queues. • **Trees:** Definitions and Concepts- Representation of binary tree, Binary tree traversals (Inorder, Postorder , preorder), • Binary search trees • **Graphs** : Representation of Graphs- Types of graphs -Breadth first traversal – Depth first traversal- - Applications of graphs –	**12**
IV	**INTRODUCTION TO ALGORITHMS:** • **INTRODUCTION**: Definition of Algorithms- Overview and importance of algorithms- pseudocode conventions, Asymptotic notations, practical complexities. • **Divide-and-Conquer:** : General Method – Binary Search- Quick Sort- Merge Sort. • **Greedy Method:** General method- Knapsack problem- Tree vertex splitting- Job sequencing with deadlines	**12**
V	**DYNAMIC PROGRAMMING, BACKTRACKING & BRANCH & BOUND** • **Dynamic programming**: General method, Multistage Graphs, All pairs shortest path, Single source shortest path. • **Backtracking:** General method, 8 Queens, Graph coloring, Hamiltonian cycle. • **Branch & Bound**: General method, Travelling salesperson problem.	**12**

Learning Resources:

- **Recommended Texts**

  1. Ellis Horowitz , Sartaj Sahni, Susan Anderson Freed, Second Edition , "Fundamentals of Data in C", Universities Press

  2. E. Horowitz, S. Sahni and S. Rajasekaran, Second Edition ,"Fundamentals of Computer Algorithms " Universities Press

- **Reference Books**

  1.Seymour Lipschutz ,"Data Structures with C", First Edition, Schaum's outline series in computers, Tata McGraw Hill.

  2. R.Krishnamoorthy and G.Indirani Kumaravel, Data Structures using C, Tata McGrawHill – 2008.

  3.A.K.Sharma, Data Structures using C , Pearson Education India,2011.

  4. G. Brassard and P. Bratley, "Fundamentals of Algorithms", PHI, New Delhi, 1997.

  5. A.V. Aho, J.E. Hopcroft, J.D. Ullmann,, "The design and analysis of Computer Algorithms", Addison Wesley, Boston, 1974

  6..Thomas H. Cormen, C.E. Leiserson, R L.Rivest and C. Stein, Introduction to Algorithms, Third edition, MIT Press, 2009

  7.Sanjoy Dasgupta, C.Papadimitriou and U.Vazirani , Algorithms , Tata McGraw-Hill, 2008.

- **Web resources:** Web resources from NDL Library, E-content from open source libraries

Mapping with Programme Outcomes:

Map course outcomes for each course with programme outcomes (PO) in the 3-point scale of Strong, Medium and Low

CO/PSO	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5	PSO 6
CO 1	2	3	3	3	3	2
CO 2	3	3	3	3	3	3
CO 3	3	3	2	3	2	3
CO 4	2	3	3	3	3	2
CO 5	3	3	2	2	3	3
Weightage of course contributedto each PSO	13	15	13	14	14	13

Course Code: 23U2CSP02	**Data Structures & Algorithms Lab**		**Credits: 4**
Lecture Hours: (L) per week	**Tutorial Hours : (T) per week**	**Lab Practice Hours: (P)per week: 5**	**Total: (L+T+P) per week: 5**
Course Category : CC3	**Year & Semester: I Year II Semester**	**Admission Year: 2023-2024**	
Pre-requisite	Basic skills in problem solving		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

- To understand and implement basic data structures using C
- To apply linear and non-linear data structures in problem solving.
- To learn to implement functions and recursive functions by means of data structures
- To implement searching and sorting algorithms

Course Outcomes: (for students: To know what they are going to learn)

CO1: Implement data structures using C

CO2: Implement various types of linked lists and their applications

CO3: Implement Tree Traversals

CO4: Implement various algorithms in C

CO5: Implement different sorting and searching algorithms

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

	List of Exercises:	**Required Hours**
	Implement the following exercises using C Programming language: 1. Array implementation of stacks 2. Array implementation of Queues 3. Linked list implementation of stacks 4. Linked list implementation of Queues 5. Binary Tree Traversals (Inorder, Preorder, Postorder) 6. Implementation of Linear search and binary search 7. Implementation Insertion sort, Quick sort and Merge Sort 8. Implementation of Depth-First Search & Breadth-First Search of Graphs. 9. Finding all pairs of Shortest Path of a Graph. 10. Finding single source shortest path of a Graph.	**60**

Learning Resources:

- **Recommended Texts**

  1. Ellis Horowitz , Sartaj Sahni, Susan Anderson Freed, Second Edition , "Fundamentals of Data in C", Universities Press

  2. E. Horowitz, S. Sahni and S. Rajasekaran, Second Edition ,"Fundamentals of Computer Algorithms " Universities Press

- **Reference Books**

  1. Seymour Lipschutz,"Data Structures with C", First Edition, Schaum's outline series in computers, Tata McGraw Hill.

  2. R.Krishnamoorthy and G.Indirani Kumaravel, Data Structures using C, Tata McGrawHill – 2008.

  3.A.K.Sharma, Data Structures using C , Pearson Education India,2011.

  4. G. Brassard and P. Bratley, "Fundamentals of Algorithms", PHI, New Delhi, 1997.

  5. A.V. Aho, J.E. Hopcroft, J.D. Ullmann,, "The design and analysis of Computer Algorithms", Addison Wesley, Boston, 1974

  6. Thomas H. Cormen, C.E. Leiserson, R L.Rivest and C. Stein, Introduction to Algorithms, Third edition, MIT Press, 2009

  7.Sanjoy Dasgupta, C.Papadimitriou and U.Vazirani , Algorithms , Tata McGraw-Hill, 2008.

- **Web resources:** Web resources from NDL Library, E-content from open source libraries

Course Code: 23U2CSS02	Advanced Excel		Credits: 2
Lecture Hours: (L) per week: 2	Tutorial Hours : (T) per week	Lab Practice Hours: (P)per week	Total: (L+T+P) per week: 2
Course Category : SEC-3	Year & Semester: I Year II Semester		Admission Year: 2023-2024
Pre-requisite	Basic knowledge in office automation / Excel		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

The objective of this course is to help the students learn the advanced features of Excel, to summarizes, analyse, explore, and present visualizations of data in the form of charts, graphs.

Course Outcomes: (for students: To know what they are going to learn)

CO1: Handle large amounts of data

CO2: Aggregate numeric data and summarise into categories and subcategories

CO3: Filtering, sorting, and grouping data or subsets of data

CO4: Create pivot tables to consolidate data from multiple files

CO5: Presenting data in the form of charts and graphs

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	Contents	Required Hours
I	Basics of Excel- Customizing common options- Absolute and relative cells- Protecting and un-protecting worksheets and cells- Working with Functions - Writing conditional expressions - logical functions - lookup and reference functions- VlookUP with Exact Match, Approximate Match- Nested VlookUP with Exact Match- VlookUP with Tables, Dynamic Ranges- Nested VlookUP with Exact Match- Using VLookUP to consolidate Data from Multiple Sheets	06
II	Data Validations - Specifying a valid range of values - Specifying a list of valid values- Specifying custom validations based on formula - Working with Templates Designing the structure of a template- templates for standardization of worksheets - Sorting and Filtering Data - Sorting tables- multiple-level sorting- custom sorting-	06

	Filtering data for selected view - advanced filter options- Working with Reports Creating subtotals- Multiple-level subtotal.	
III	Creating Pivot tables Formatting and customizing Pivot tables- advanced options of Pivot tables- Pivot charts- Consolidating data from multiple sheets and files using Pivot tables- external data sources- data consolidation feature to consolidate data- Show Value As % of Row, % of Column, Running Total,Compare with Specific Field- Viewing Subtotal under Pivot- Creating Slicers.	**06**
IV	More Functions Date and time functions- Text functions- Database functions- Power Functions - Formatting Using auto formatting option for worksheets- Using conditional formatting option for rows, columns and cells- WhatIf Analysis - Goal Seek- Data Tables- Scenario Manager.	**06**
V	Charts - Formatting Charts- 3D Graphs- Bar and Line Chart together- Secondary Axis in Graphs- Sharing Charts with PowerPoint / MS Word, Dynamically- New Features Of Excel Spark lines, Inline Charts, data Charts- Overview of all the new features.	**06**

Learning Resources:

- **Recommended Tex**

  Excel 2019 All-in-One For Dummies – 2018- Greg Harvey

- **Reference Books**

  Microsoft Excel 2019 Pivot Table Data Crunching-2019,Bill Jelen and Michael Alexander

- **Web resources:** Web resources from NDL Library, E-content from open source libraries

Mapping with Programme Outcomes:

Map course outcomes for each course with programme outcomes (PO) in the 3-point scale of Strong, Medium and Low

CO/PSO	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5	PSO 6
CO 1	2	3	2	2	3	3
CO 2	3	3	3	3	3	3
CO 3	3	3	3	3	3	3
CO 4	3	3	2	3	3	3
CO 5	3	3	3	2	2	3
Weightage of course contributedto each PSO	14	15	13	13	14	15

Course Code: 23U3CSC03	Microprocessor and Microcontroller		Credits: 4
Lecture Hours: (L) per week: 4	Tutorial Hours : (T) per week	Lab Practice Hours: (P)per week	Total: (L+T+P) per week: 4
Course Category :CC5	Year & Semester: II Year III Semester		Admission Year: 2023-2024
Pre-requisite	Basic knowledge on micro processor and micro controllers		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

- To introduce the internal organization of Intel 8085 Microprocessor.
- To enable the students to write assembly language programs using 8085.
- To interface the peripheral devices to 8085 using Interrrupt controller and DMA interface.
- To provide real-life applications using microcontroller.

Course Outcomes: (for students: To know what they are going to learn)

CO1: Remember the Basic binary codes and their conversions. Binary concepts are used in Microprocessor programming and provide a good understanding of the architecture of 8085.

CO2: Understanding the 8085 instruction set and their classifications, enables the students to write the programs easily on their own using different logic..

CO3: Applying different types of instructions to convert binary codes and analyzing the outcome. The instruction set is applied to develop programs on multibyte arithmetic operations.

CO4: Analyze how peripheral devices are connected to 8085 using Interrupts and DMA controller.

CO5: An exposure to create real time applications using microcontroller.

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	Contents	Required Hours
I	Digital Computers - Microcomputer Organization-Computer languages –Microprocessor Architecture and its operations – Microprocessor initiated operations and 8085 Bus organization – Internal Data operations and 8085 registers - Peripheral or External initiated operations.	12
II	8085 Microprocessor – Pinout and Signals – Functional block diagram - 8085 Instruction Set and Classifications.	12
III	BCD to Binary and Binary to BCD conversions - ASCII to BCD and BCD to ASCII conversions - Binary to ASCII and ASCII to Binary conversions. BCD Arithmetic - BCD	12

	addition and Subtraction - Multibyte Addition and Subtraction - Multiplication and Division.	
IV	The 8085 Interrupts – RIM AND SIM instructions-8259 Programmable Interrupt Controller-Direct Memory Access (DMA) and 8257 DMA controller.	**12**
V	Introduction to Microcontroller - Microcontroller Vs Microprocessor - 8051 Microcontroller architecture - 8051 pin description. Timers and Counters – Operating Modes-Control Registers. Interrupts – Interrupts in 8051 - Interrupts Control Register – Execution of interrupt.	**12**

Learning Resources:

- **Recommended Texts**

1. R. S. Gaonkar- "Microprocessor Architecture- Programming and Applications with 8085"- 5th Edition- Penram International Publications,2009. [For unit I to unit IV].

2. Soumitra Kumar Mandal -"Microprocessors and Microcontrollers – Architectures, Programming and Interfacing using 8085, 8086, 8051", Tata McGraw Hill Education Private Limited. [for unit V].

- **Reference Books**

1. Mathur- "Introduction to Microprocessor"- 3rd Edition- Tata McGraw-Hill -1993.
2. Raj Kamal - "Microcontrollers: Architecture, Programming, Interfacing and System Design", Pearson Education, 2005.
3. Krishna Kant, "Microprocessors and Microcontrollers – Architectures, Programming and System Design 8085, 8086, 8051, 8096", PHI, 2008.

- **Web resources:** Web resources from NDL Library, E-content from open source libraries

Mapping with Programme Outcomes:

Map course outcomes for each course with programme outcomes (PO) in the 3-point scale of Strong, Medium and Low

CO/PSO	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5	PSO 6
CO 1	3	3	2	3	3	3
CO 2	3	3	3	3	3	2
CO 3	3	2	3	3	2	3
CO 4	3	3	3	3	3	2
CO 5	3	2	3	3	3	3
Weightage of course contributedto each PSO	15	13	14	15	14	13

Course Code: 23U3CSP03	Microprocessor and Microcontroller Lab		Credits: 4
Lecture Hours: (L) per week:4	Tutorial Hours : (T) per week	Lab Practice Hours: (P)per week: 4	Total: (L+T+P) per week: 4
Course Category :CC6	Year & Semester: II Year III Semester		Admission Year: 2023-2024
Pre-requisite	Basic knowledge on 8085 micro processor and micro controllers		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

- To introduce the internal organization of Intel 8085 Microprocessor.
- To enable the students to write assembly language programs using 8085.
- To interface the peripheral devices to 8085 using Interrupt controller and DMA interface.
- To provide real-life applications using microcontroller.

Course Outcomes: (for students: To know what they are going to learn)

CO1: Remember the Basic binary codes and their conversions. Binary concepts are used in Microprocessor programming and provide a good understanding of the architecture of 8085.

CO2: Understanding the 8085-instruction set and their classifications, enables the students to write the programs easily on their own using different logic.

CO3: Applying different types of instructions to convert binary codes and analyzing the outcome. The instruction set is applied to develop programs on multibyte arithmetic operations.

CO4: Analyze how peripheral devices are connected to 8085 using Interrupts and DMA controller.

CO5: An exposure to create real time applications using microcontroller.

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

	List of Exercises:	Required Hours
	Addition and Subtraction 1. 8 - bit addition 2. 16 - bit addition 3. 8 - bit subtraction 4. BCD subtraction II. Multiplication and Division 1. 8 - bit multiplication	

	2. BCD multiplication 3. 8 - bit division 4. III. Sorting and Searching 1. Searching for an element in an array. 2. Sorting in Ascending and Descending order. 3. Finding the largest and smallest elements in an array. 4. Reversing array elements. 5. Block move. IV. Code Conversion 1. BCD to Hex and Hex to BCD 2. Binary to ASCII and ASCII to binary 3. ASCII to BCD and BCD to ASCII V. Simple programs on 8051 Microcontroller 1. Addition 2. Subtraction 3. Multiplication 4. Division 5. Interfacing Experiments using 8051 I. Realisation of Boolean Expression through ports. II. Time delay generation using subroutines. III. Display LEDs through ports	**60**

Learning Resources:

- **Recommended Texts**

1. R. S. Gaonkar- "Microprocessor Architecture- Programming and Applications with 8085"- 5th Edition- Penram International Publications,2009. [For unit I to unit IV].

2. Soumitra Kumar Mandal -"Microprocessors and Microcontrollers – Architectures, Programming and Interfacing using 8085, 8086, 8051", Tata McGraw Hill Education Private Limited. [for unit V].

- **Reference Books**

3. Mathur- "Introduction to Microprocessor"- 3rd Edition- Tata McGraw-Hill -1993.

4. Raj Kamal - "Microcontrollers: Architecture, Programming, Interfacing and System Design", Pearson Education, 2005.

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

Web resources: Web resources from NDL Library, E-content from open-source libraries

Course Code: 23U3CSGE03	DISCRETE MATHEMATICAL STRUCTURES		Credits: 3
Lecture Hours: (L) per week: 4	**Tutorial Hours : (T) per week**	**Lab Practice Hours: (P)per week**	**Total: (L+T+P) per week: 4**
Course Category :EC-3	**Year & Semester: II Year III Semester**		**Admission Year: 2023-2024**
Pre-requisite	Basic Knowledge on probability and mathematical logic		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

To understand the mathematical concepts like set theory, logics, number theory, combinatory and relations.

Course Outcomes: (for students: To know what they are going to learn)

CO1: To gain knowledge on set theory

CO2: Able to understand different mathematical logics and functions

CO3: To get an idea on Permutations and Combinations

CO4: Understanding the different form of number theory

CO5: Able to understand Relations and its applications .

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	Contents	Required Hours
I	***SET THEORY*** Introduction- set and Its Element – Set Description (Roster, Set Builder and cardinal number method) Types of Sets- Set Operations and Laws of set Theory. Partition of sets. Minsets-Countable and un Countable set. Algebra of sets and Duality	**12**
II	***MATHEMATICAL LOGIC*** Basic Logic and Proof, logical operations – Logic Propositional equivalence, Predicates and Quantities, Tautology-Contradiction-Methods of proofs(Direct and Indirect)- Function- Definition-Notation- Types of Function- Composition of Functions-	**12**
III	***NUMBER THEORY***	

	The Integers and Division, Integers and Algorithms,(Multiplication, Addition and Division -Sequences and Summations, Recursive algorithms, Program correctness	**12**
IV	***COMBINATORICS*:** The basics of counting, the pigeonhole principle, Permutations and Combinations, Binomial coefficients, Generalized permutations and combinations	**12**
V	***RELATIONS*** Relations – Relations and their properties, Representing Relations, Closures of relations, Equivalence relations, Partial orderings-Recurrence Relations Binary Relations.	**12**

Learning Resources:

- **Recommended Texts**

1. Rosen K.H. Discrete Mathematics and its Applications, 5th edition, Tata McGraw – Hills, 2003.
2. J.K Sharma "DISCRETE MATHEMATICS" 3 rd Edition Macmillan Reprint 2011

Reference Books

1. Johnson Baugh R, and Carman R, Discrete mathematics, 5th edition, Person Education, 2003.
2. Kolman B, Busoy R.C, and Ross S.C, Discrete Mathematical Structures, 5th edition, Pretitice – Hall, 2004.
3. Mott J.L, Kandel A, and Bake T.P, Discrete Mathematics for Computer Scientists & Mathematicians, 2nd edition, Prentice-Hall of India, 2002.

- **Web resources:** Web resources from NDL Library, E-content from open-source libraries

Mapping with Programme Outcomes:

Map course outcomes for each course with programme outcomes (PO) in the 3-point scale of Strong, Medium and Low

CO/PSO	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5	PSO 6
CO 1	3	3	2	3	3	3
CO 2	2	3	3	3	3	3
CO 3	3	2	3	3	2	3
CO 4	3	3	3	3	3	2
CO 5	2	3	3	3	2	3
Weightage of course contributedto each PSO	13	14	14	15	13	14

Course Code: 23U3CSS03	**PHP Programming**		**Credits: 02**
Lecture Hours: (L) per week: 4	**Tutorial Hours : (T) per week**	**Lab Practice Hours: (P)per week**	**Total: (L+T+P) per week: 4**
Course Category :	**Year & Semester: II Year III Semester**		**Admission Year: 2023-2024**
Pre-requisite	Basic Knowledge on Web		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

The objective of this course is to teach the fundamentals of quantum information processing, including quantum computation, quantum cryptography, and quantum information theory.

Course Outcomes: (for students: To know what they are going to learn)

CO1: Analyze the behaviour of basic quantum algorithms

CO2: Implement simple quantum algorithms and information channels in the quantum circuit model

CO3: Simulate a simple quantum error-correcting code

CO4: Prove basic facts about quantum information channels

CO5:

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	Contents	Required Hours
I	Introduction to PHP -Basic Knowledge of websites -Introduction of Dynamic Website -Introduction to PHP -Scope of PHP -XAMPP and WAMP Installation- PHP Programming Basics -Syntax of PHP -Embedding PHP in HTML -Embedding HTML in PHP .	**12**
II	Introduction to PHP Variable -Understanding Data Types -Using Operators -Using Conditional Statements -If(), else if() and else if condition Statement -Switch() Statements -Using the while() Loop -Using the for() Loop	**12**
III	PHP Functions -PHP Functions -Creating an Array -Modifying Array Elements -Processing Arrays with Loops -Grouping Form Selections with Arrays -Using Array Functions -Using Predefined PHP Functions -Creating User-	**12**

	Defined Functions	
IV	PHP Advanced Concepts -Reading and Writing Files - Reading Data from a File -Managing Sessions and Using Session Variables -Destroying a Session -Storing Data in Cookies -Setting Cookies	**12**
V	OOPS Using PHP -OOPS Concept-Class, Object, Abstractions, Encapsulation, Inheritance, Polymorphism - Creating Classes and Object in PHP-Cookies and Session Management-Working with forms and system file - Error Handling- Model View Controller – AJAX.	**12**

Learning Resources:

- **Recommended Texts**

  Head First PHP & MySQL: A Brain-Friendly Guide- 2009-Lynn mighley and Michael Morrison.

- **Reference Books**

  The Joy of PHP: A Beginner's Guide to Programming Interactive Web Applications with PHP and MySQL- Alan Forbes

Web resources: Web resources from NDL Library, E-content from open-source libraries

Mapping with Programme Outcomes:

Map course outcomes for each course with programme outcomes (PO) in the 3-point scale of Strong, Medium and Low

CO/PSO	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5	PSO 6
CO 1	3	3	3	3	3	3
CO 2	2	3	3	3	3	3
CO 3	3	2	3	3	3	2
CO 4	3	3	3	3	3	2
CO 5	3	2	3	3	2	3
Weightage of course contributedto each PSO	14	13	15	15	14	13

Course Code: 23U4CSC04	Java Programming		Credits: 4
Lecture Hours: (L) per week: 4	Tutorial Hours : (T) per week	Lab Practice Hours: (P)per week	Total: (L+T+P) per week: 4
Course Category :CORE	Year & Semester: II Year IV Semester		Admission Year: 2023-2024
Pre-requisite	Basic Programming skill		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

- To provide fundamental knowledge of object-oriented programming.
- To equip the student with programming knowledge in Core Java from the basics up.
- To enable the students to use AWT controls, Event Handling and Swing for GUI.

Course Outcomes: (for students: To know what they are going to learn)

CO1: Understand the basic Object-oriented concepts.Implement the basic constructs of Core Java.**CO2:**Implement inheritance, packages, interfaces and exception handling of Core Java.

CO3: Implement multi-threading and I/O Streams of Core Java

CO4: Implement AWT and Event handling.

CO5: Use Swing to create GUI.

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	Contents	Required Hours
I	**Introduction:** Review of Object Oriented concepts - History of Java - Java buzzwords - JVM architecture - Data types - Variables - Scope and life time of variables - arrays - operators - control statements - type conversion and casting - simple java program - constructors - methods - Static block - Static Data - Static Method String and String Buffer Classes	12
II	**Inheritance**: Basic concepts - Types of inheritance - Member access rules - Usage of this and Super key word - Method Overloading - Method overriding - Abstract classes - Dynamic method dispatch - Usage of final keyword. **Packages**: Definition - Access Protection - Importing Packages. **Interfaces**: Definition – Implementation – Extending Interfaces. **Exception Handling**: *try – catch - throw - throws – finally* – Built-inexceptions - Creating own Exception classes.	12

III	**Multithreaded Programming**: Thread Class - Runnable interface – Synchronization – Using synchronized methods – Using *synchronized* statement - Interthread Communication – Deadlock. **I/O Streams:** Concepts of streams - Stream classes- Byte and Character stream - Reading console Input and Writing Console output - File Handling.	**12**
IV	**AWT Controls:** The AWT class hierarchy - user interface components- Labels - Button - Text Components - Check Box - Check Box Group - Choice - List Box - Panels – Scroll Pane - Menu - Scroll Bar. Working with Frame class - Colour - Fonts and layout managers. **Event Handling:** Events - Event sources - Event Listeners - Event Delegation Model (EDM) - Handling Mouse and Keyboard Events - Adapter classes - Inner classes.	**12**
V	**Swing:** Introduction to Swing - Hierarchy of swing components. Containers - Top level containers - JFrame - JWindow - JDialog - JPanel - JButton - JToggleButton - JCheckBox - JRadioButton - JLabel,JTextField - JTextArea - JList - JComboBox - JScrollPane	**12**

Learning Resources:

- **Recommended Texts**

1. Herbert Schildt, The Complete Reference, Tata McGraw Hill, New Delhi, 7th Edition, 2010.
2. Gary Cornell, Core Java 2 Volume I – Fundamentals, Addison Wesley, 1999.

Reference Books

1. Head First Java, O'Rielly Publications,
2. Y. Daniel Liang, *Introduction to Java Programming*, 7th Edition, Pearson Education India, 2010.

Web resources: Web resources from NDL Library, E-content from open-source libraries

Mapping with Programme Outcomes:

Map course outcomes for each course with programme outcomes (PO) in the 3-point scale of Strong, Medium and Low

CO/PSO	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5	PSO 6
CO 1	3	3	2	3	3	2
CO 2	2	3	2	3	3	3
CO 3	3	2	3	3	2	3
CO 4	3	3	3	3	3	2
CO 5	3	3	3	3	3	3
Weightage of course contributedto each PSO	14	15	13	15	14	13

Course Code: 23U4CSP04	Java Programming Lab		Credits: 4
Lecture Hours: (L) per week:04	**Tutorial Hours : (T) per week**	**Lab Practice Hours: (P)per week**	**Total: (L+T+P) per week: 4**
Course Category :	**Year & Semester: II Year IV Semester**		**Admission Year: 2023-2024**
Pre-requisite	Basic Programming debugging skills		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

- To gain practical expertise in coding Core Java programs
- To become proficient in the use of AWT, Event Handling and Swing.

Course Outcomes: (for students: To know what they are going to learn)

CO1: Code, debug and execute Java programs to solve the given problems

CO2: Implement multi-threading and exception-handling

CO3: Implement functionality using String and StringBuffer classes

CO4: Demonstrate Event Handling.

CO5: Create applications using Swing and AWT

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

	List of Exercises:	Required Hours
	1. Write a Java program that prompts the user for an integer and then prints out all the prime numbers up to that Integer? 2. Write a Java program to multiply two given matrices. 3. Write a Java program that displays the number of characters, lines and words in a text? 4. Generate random numbers between two given limits using Random class and print messages according to the range of the value generated. 5. Write a program to do String Manipulation using Character Array and perform the following string	

	operations: a. String length b. Finding a character at a particular position c. Concatenating two strings 6. Write a program to perform the following string operations using String class: a. String Concatenation b. Search a substring c. To extract substring from given string 7. Write a program to perform string operations using StringBuffer class: a. Length of a string b. Reverse a string c. Delete a substring from the given string 8. Write a java program that implements a multi-thread application that has three threads. First thread generates random integer every 1 second and if the value is even, second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of cube of the number. 9. Write a threading program which uses the same method asynchronouslyto print the numbers 1 to 10 using Thread1 and to print 90 to 100 using Thread2. 10. Write a program to demonstrate the use of following exceptions. a. Arithmetic Exception b. Number Format Exception c. Array Index Out of Bound Exception d. Negative Array Size Exception 11. Write a Java program that reads on file name from the user, then displays information about whether the file exists, whether the file is readable, whether the file is writable, the type of file and the length of the file in bytes? 12. Write a program to accept a text and change its size and font. Include bold italic options. Use frames and controls. 13. Write a Java program that handles all mouse events and shows the event name at the center of the window when	**60**

	a mouse event is fired. (Use adapter classes). 14. Write a Java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +, -,*, % operations. Add a text field to display the result. Handle any possible exceptions like divide by zero. 15. Write a Java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green with radio buttons. On selecting a button, an appropriate message with "stop" or "ready" or "go" should appear above the buttons in a selected color. Initially there is no message shown.	

Learning Resources:

Learning Resources:

- **Recommended Texts**

3. Herbert Schildt, The Complete Reference, Tata McGraw Hill, New Delhi, 7th Edition, 2010.
4. Gary Cornell, Core Java 2 Volume I – Fundamentals, Addison Wesley, 1999.

- **Reference Books**

3. Head First Java, O'Rielly Publications,
4. Y. Daniel Liang, *Introduction to Java Programming*, 7th Edition, Pearson Education India, 2010.

Web resources: Web resources from NDL Library, E-content from open-source libraries

Course Code: 23U4CSS04	Cloud Computing		Credits: 2
Lecture Hours: (L) per week	Tutorial Hours : (T) per week	Lab Practice Hours: (P)per week	Total: (L+T+P) per week: 2
Course Category :SEC-5	Year & Semester: II Year IV Semester	Admission Year:	
Pre-requisite	Basic knowledge on virtual storage or cloud concept		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

- To impart fundamental concepts of Cloud Computing.
- To impart a working knowledge of the various cloud service types and their uses and pitfalls.
- To enable the students to know the common features and differences in the service offerings of the three major Cloud Computing service providers, namely Amazon, Microsoft and Google.
- To provide know-how of the various aspects of application design, benchmarking and security on the Cloud.

Course Outcomes: (for students: To know what they are going to learn)

CO1: To understand the concepts and technologies involved in Cloud Computing.

CO2: To understand the concepts of various cloud services and their implementation in the Amazon, Microsoft and Google cloud computing platforms.

CO3: To understand the aspects of application design for the Cloud.

CO4: To understand the concepts involved in benchmarking and security on the Cloud.

CO5: To understand the way in which the cloud is used in various domains.

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	Contents	Required Hours
I	**Introduction to Cloud Computing:** Definition of Cloud Computing – Characteristics of Cloud Computing – Cloud Models – Cloud Service Examples – Cloud-based Services and Applications. **Cloud Concepts and Technologies:** Virtualization – Load balancing – Scalability and Elasticity – Deployment – Replication – Monitoring – Software Defined Networking – Network Function Virtualization – MapReduce – Identity and Access Management – Service Level Agreements – Billing.	06
II	**Cloud Services** **Compute Services:** Amazon Elastic Computer Cloud -	06

	Google Compute Engine - Windows Azure Virtual Machines. **Storage Services:** Amazon Simple Storage Service - Google Cloud Storage - Windows Azure Storage **Database Services:** Amazon Relational Data Store - Amazon Dynamo DB - Google Cloud SQL - Google Cloud Data Store - Windows Azure SQL Database - Windows Azure Table Service **Application Services:** Application Runtimes and Frameworks - Queuing Services - Email Services - Notifiction Services - Media Services **Open Source Private Cloud Software:** CloudStack – Eucalyptus - OpenStack	
III	**Cloud Application Design:** Introduction – Design Consideration for Cloud Applications – Scalability – Reliability and Availability – Security – Maintenance and Upgradation – Performance – Reference Architectures for Cloud Applications – Cloud Application Design Methodologies: Service Oriented Architecture (SOA), Cloud Component Model, IaaS, PaaS and SaaS Services for Cloud Applications, Model View Controller (MVC), RESTful Web Services – Data Storage Approaches: Relational Approach (SQL), Non-Relational Approach (NoSQL).	**06**
IV	**Cloud Application Benchmarking and Tuning:** Introduction to Benchmarking – Steps in Benchmarking – Workload Characteristics – Application Performance Metrics – Design Consideration for Benchmarking Methodology – Benchmarking Tools and Types of Tests – Deployment Prototyping. **Cloud Security:** Introduction – CSA Cloud Security Architecture – Authentication (SSO) – Authorization – Identity and Access Management – Data Security : Securing data at rest, securing data in motion – Key Management – Auditing.	**06**
V	**Case Studies:** Cloud Computing for Healthcare – Cloud Computing for Energy Systems - Cloud Computing for Transportation Systems - Cloud Computing for Manufacturing Industry - Cloud Computing for Education.	**06**

Learning Resources:

- **Recommended Texts**

1. Arshdeep Bahga, Vijay Madisetti, *Cloud Computing – A Hands On Approach*, Universities Press (India) Pvt. Ltd., 2018.

- **Reference Books**

1. Anthony T Velte, Toby J Velte, Robert Elsenpeter, *Cloud Computing: A Practical Approach*, Tata McGraw-Hill, 2013.
2. Barrie Sosinsky, *Cloud Computing Bible*, Wiley India Pvt. Ltd., 2013.
3. David Crookes, *Cloud Computing in Easy Steps*, Tata McGraw Hill, 2012.
4. Dr. Kumar Saurabh, *Cloud Computing*, Wiley India, Second Edition 2012.

Web resources: Web resources from NDL Library, E-content from open-source libraries

Mapping with Programme Outcomes:

Map course outcomes for each course with programme outcomes (PO) in the 3-point scale of Strong, Medium and Low

CO/PSO	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5	PSO 6
CO 1	3	3	3	3	3	3
CO 2	2	3	3	3	3	3
CO 3	3	2	3	3	3	3
CO 4	3	3	3	3	3	2
CO 5	3	3	3	2	2	3
Weightage of course contributedto each PSO	14	14	15	14	14	14

Course Code: 23U4CSS05	Big Data Analytics		Credits: 2
Lecture Hours: (L) per week	**Tutorial Hours : (T) per week**	**Lab Practice Hours: (P)per week**	**Total: (L+T+P) per week: 2**
Course Category :SEC-7	**Year & Semester:II Year IV Semester**	**Admission Year:**	
Pre-requisite	Basic knowledge on Data handlings		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

1. To know the fundamental concepts of big data and analytics.
2. To explore tools and practices for working with big data.

Course Outcomes: (for students: To know what they are going to learn)

CO1: Work with big data tools and its analysis techniques.

CO2: Analyze data by utilizing clustering and classification algorithms.

CO3: Learn and apply different mining algorithms and recommendation systems for large volumes of data.

CO4: Perform analytics on data streams.

CO5: Learn NoSQL databases and management.

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	Contents	Required Hours
I	**INTRODUCTION TO BIG DATA :** Evolution of Big data — Best Practices for Big data Analytics — Big data characteristics — Validating — The Promotion of the Value of Big Data — Big Data Use Cases- Characteristics of Big Data Applications — Perception and Quantification of Value -Understanding Big Data Storage — A General Overview of High-Performance Architecture — HDFS — MapReduce and YARN — Map Reduce Programming Model	**06**
II	**CLUSTERING AND CLASSIFICATION :**Advanced Analytical Theory and Methods: Overview of Clustering — K-means — Use Cases — Overview of the Method — Determining the Number of Clusters — Diagnostics — Reasons to Choose and Cautions .- Classification: Decision Trees — Overview of a Decision Tree — The General Algorithm — Decision Tree Algorithms — Evaluating a Decision Tree — Decision Trees in R — Naïve Bayes — Bayes? Theorem — Naïve Bayes Classifier	**06**

III	**ASSOCIATION AND RECOMMENDATION SYSTEM:**Advanced Analytical Theory and Methods: Association Rules — Overview — Apriori Algorithm — Evaluation of Candidate Rules — Applications of Association Rules — Finding Association& finding similarity — Recommendation System: Collaborative Recommendation- Content Based Recommendation — Knowledge Based Recommendation- Hybrid Recommendation Approaches	**06**
IV	**STREAM MEMORY:** Introduction to Streams Concepts — Stream Data Model and Architecture — Stream Computing, Sampling Data in a Stream — Filtering Streams — Counting Distinct Elements in a Stream — Estimating moments — Counting oneness in a Window — Decaying Window — Real time Analytics Platform(RTAP) applications — Case Studies — Real Time Sentiment Analysis, Stock Market Predictions. Using Graph Analytics for Big Data: Graph Analytics	**06**
V	**NOSQL DATA MANAGEMENT FOR BIG DATA AND VISUALIZATION :**NoSQL Databases : Schema-less Models?: Increasing Flexibility for Data Manipulation-Key Value Stores- Document Stores — Tabular Stores — Object Data Stores — Graph Databases Hive — Sharding —Hbase — Analyzing big data with twitter — Big data for E-Commerce Big data for blogs — Review of Basic Data Analytic Methods using R.	**06**

Learning Resources:

- **Recommended Texts**

1. Anand Rajaraman and Jeffrey David Ullman, "Mining of Massive Datasets", Cambridge University Press, 2012.

- **Reference Books**

1. David Loshin, "Big Data Analytics: From Strategic Planning to Enterprise Integration with Tools, Techniques, NoSQL, and Graph", Morgan Kaufmann/El sevier Publishers, 2013.

2. 2. EMC Education Services, "Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data", Wiley publishers, 2015.

Web resources: Web resources from NDL Library, E-content from open-source libraries

Mapping with Programme Outcomes:

Map course outcomes for each course with programme outcomes (PO) in the 3-point scale of Strong, Medium and Low

CO/PSO	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5	PSO 6
CO 1	3	3	3	3	3	3
CO 2	3	3	3	2	3	2
CO 3	3	2	3	3	3	3
CO 4	3	3	3	3	3	2
CO 5	3	3	2	2	3	3
Weightage of course contributedto each PSO	15	14	14	13	15	13

Course Code: 23U5CSC05	Software Engineering		Credits: 4
Lecture Hours: (L) per week: 5	Tutorial Hours : (T) per week	Lab Practice Hours: (P)per week	Total: (L+T+P) per week: 5
Course Category :CC9	Year & Semester: III Year V Semester		Admission Year: 2023-2024
Pre-requisite	Basic Knowledge on Software Applications		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

- To understand the software engineering concepts and to create a system model in real life applications

Course Outcomes: (for students: To know what they are going to learn)

CO1: Gain basic knowledge of analysis and design of systems

CO2: Ability to apply software engineering principles and techniques

CO3: Model a reliable and cost-effective software system

CO4: Ability to design an effective model of the system

CO5: Perform Testing at various levels and produce an efficient system.

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	Contents	Required Hours
I	Introduction: The software engineering discipline, programs vs. software products, why study software engineering, emergence of software engineering, Notable changes in software development practices, computer systems engineering. Software Life Cycle Models: Why use a life cycle model, Classical waterfall model, iterative waterfall model, prototyping model, evolutionary model, spiral model, comparison of different life cycle models.	12
II	**Requirements Analysis and Specification:** Requirements gathering and analysis, Software requirements specification (SRS) **Software Design**: Good software design, cohesion and coupling, neat arrangement, software design approaches,	12

	object- oriented vs function-oriented design	
III	Function-Oriented Software Design: Overview of SA/SD methodology, structured analysis, data flow diagrams (DFD's), structured design, detailed design. User-Interface design: Characteristics of a good interface; basic concepts; types of user interfaces; component based GUI development, a user interface methodology.	**12**
IV	Coding and Testing: Coding; code review; testing; testing in the large vs testing in the small; unit testing; black-box testing; white-box testing; debugging; program analysis tools; integration testing; system testing; some general issues associated with testing. Software Reliability and Quality Management: Software reliability; statistical testing; software quality; software quality management system; SEI capability maturity model; personal software process.	**12**
V	Computer Aided Software Engineering: CASE and its scope; CASE environment; CASE support in software life cycle; other characteristics of CASE tools; towards second generation CASE tool; architecture of a CASE environment. Software Maintenance: Characteristic of software maintenance; software reverse engineering; software maintenance process models; estimation of maintenance cost;	**12**

Learning Resources:

- **Recommended Texts**

1. Rajib Mall, Fundamentals of Software Engineering, Fifth Edition, Prentice-Hall of India, 2018

- **Reference Books**

1. Richard Fairley, Software Engineering Concepts, Tata McGraw-Hill publishing company Ltd, Edition 1997.
2. Roger S. Pressman, Software Engineering, Seventh Edition, McGraw-Hill.

James A. Senn, Analysis & Design of Information Systems, Second Edition, McGraw-Hill International Editions.

Web resources: Web resources from NDL Library, E-content from open-source libraries

Mapping with Programme Outcomes:

Map course outcomes for each course with programme outcomes (PO) in the 3-point scale of Strong, Medium and Low

CO/PSO	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5	PSO 6
CO 1	3	3	3	3	3	3
CO 2	2	3	3	3	3	3
CO 3	2	2	3	3	3	3
CO 4	3	3	3	2	3	3
CO 5	3	3	3	2	3	3
Weightage of course contributedto each PSO	13	14	15	13	15	15

Course Code: 23U5CSC06	Database Management Systems		Credits:4
Lecture Hours: (L) per week: 5	**Tutorial Hours : (T) per week**	**Lab Practice Hours: (P)per week**	**Total: (L+T+P) per week: 5**
Course Category :CC-10	**Year & Semester: III YEAR V SEMESTER**		**Admission Year: 2023-2024**
Pre-requisite	Basic knowledge on Data and its relations		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

- To enable the students to learn the designing of data base systems, foundation on the relational model of data and normal forms.
- To understood the concepts of data base management system, design simple Database models
- To learn and understand to write queries using SQL, PL/SQL.

Course Outcomes: (for students: To know what they are going to learn)

CO1: Understand the various basic concepts of Data Base System. Difference between file system and DBMS and compare various data models.

CO2: Define the integrity constraints. Understand the basic concepts of Relational Data Model, Entity-Relationship Model.

CO3: Design database schema considering normalization and relationships within database. Understand and construct database using Structured Query Language. Attain a good practical skill of managing and retrieving of data using Data Manipulation Language (DML).

CO4: Classify the different functions and various join operations and enhance the knowledge of handling multiple tables.

CO5: Learn to design Data base operations and implement using PL/SQL programs. Learn basics of PL/SQL and develop programs using Cursors, Exceptions

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	Contents	Required Hours
I	**Database Concepts:**Database Systems - Data vs Information - Introducing the database -File system - Problems with file system – Database systems. Data models - Importance - Basic Building Blocks - Business rules - Evolution of Data models - Degrees of Data Abstraction	**12**
II	**Design Concepts:** Relational database model - logical view of data-keys -Integrity rules - relational set operators - data dictionary and the system catalog - relationships -data redundancy revisited -indexes - codd's rules. Entity	**12**

	relationship model - ER diagram	
III	**Normalization of Database Tables**: Database tables and Normalization – The Need for Normalization –The Normalization Process – Higher level Normal Form. **Introduction to SQL**: Data Definition Commands – Data Manipulation Commands – SELECT Queries – Additional Data Definition Commands – Additional SELECT Query Keywords – Joining Database Tables.	**12**
IV	**Advanced SQL**:Relational SET Operators: UNION – UNION ALL – INTERSECT - MINUS.SQL Join Operators: Cross Join – Natural Join – Join USING Clause – JOIN ON Clause – Outer Join.**Sub Queries and Correlated Queries**: WHERE – IN – HAVING – ANY and ALL – FROM. SQL Functions: Date and Time Function – Numeric Function – String Function – Conversion Function	**12**
V	**PL/SQL**:A Programming Language: History – Fundamentals – Block Structure – Comments – Data Types – Other Data Types – Variable Declaration – Assignment operation – Arithmetic operators.**Control Structures and Embedded SQL**: Control Structures – Nested Blocks – SQL in PL/SQL – Data Manipulation – Transaction Control statements. **PL/SQL Cursors and Exceptions**: Cursors – Implicit Cursors, Explicit Cursors and Attributes – Cursor FOR loops – SELECT…FOR UPDATE – WHERE CURRENT OF clause – Cursor with Parameters – Cursor Variables – Exceptions – Types of Exceptions.	**12**

Learning Resources:

- **Recommended Texts**
  1. Coronel, Morris, Rob, "Database Systems, Design, Implementation and Management", Ninth Edition
  2. Nilesh Shah, "Database Systems Using Oracle", 2nd edition, Pearson Education India, 2016
- **Reference Books**

1.Abraham Silberschatz, Henry F.Korth and S.Sudarshan,“Database System Concepts”, McGraw Hill International Publication ,VI Edition.

2.Shio Kumar Singh , “Database Systems “,Pearson publications ,II Edition

Web resources: Web resources from NDL Library, E-content from open-source libraries	**PSO 1**	**PSO 2**	**PSO 3**	**PSO 4**	**PSO 5**	**PSO 6**
CO 1	3	3	3	3	3	3
CO 2	3	3	3	3	3	3
CO 3	3	2	3	2	2	3
CO 4	3	2	3	2	3	3
CO 5	3	3	3	3	2	3
Weightage of course contributedto each PSO	15	13	15	13	13	15

Course Code: 23U5CSP05	**DATABASE MANAGEMENT SYSTEMS LAB**		**Credits:4**
Lecture Hours: (L) per week	**Tutorial Hours : (T) per week**	**Lab Practice Hours: (P)per week: 5**	**Total: (L+T+P) per week:5**
Course Category :CC-11	**Year & Semester: III Year V semester**		**Admission Year: 2023-2024**
Pre-requisite	Basic Knowledge on Database Tools		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

Students can learn various SQL and PL/SQL commands, cursor and various application programs.

Course Outcomes: (for students: To know what they are going to learn)

CO1: Understand the various basic concepts of Data Base System. Difference between file system and DBMS and compare various data models.

CO2: Define the integrity constraints. Understand the basic concepts of Relational Data Model, Entity-Relationship Model.

CO3: Design database schema considering normalization and relationships within database. Understand and construct database using Structured Query Language. Attain a good practical skill of managing and retrieving of data using Data Manipulation Language (DML).

CO4: Classify the different functions and various join operations and enhance the knowledge of handling multiple tables.

CO5: Learn to design Data base operations and implement using PL/SQL programs. Learn basics of PL/SQL and develop programs using Cursors, Exceptions

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

	List of Exercises:	**Required Hours**
	I. SQL 1. DDL COMMANDS	

	2. DML COMMANDS 3. TCL COMMANDS ***II. PL/SQL*** 4. FIBONACCI SERIES 5. FACTORIAL 6. STRING REVERSE 7. SUM OF SERIES 8. TRIGGER ***III. CURSOR*** 9. STUDENT MARK ANALYSIS USING CURSOR ***IV. APPLICATION*** 10. LIBRARY MANAGEMENT SYSTEM 11. STUDENT MARK ANALYSIS	

Learning Resources:

- **Recommended Texts**

1 Coronel, Morris, Rob, "Database Systems, Design, Implementation and Management", Ninth Edition

2 Nilesh Shah, "Database Systems Using Oracle", 2nd edition, Pearson Education India, 2016

- **Reference Books**

1 . Abraham Silberschatz, Henry F.Korth and S.Sudarshan,"Database System Concepts", McGraw Hill International Publication ,VI Edition.

2. Shio Kumar Singh , "Database Systems ",Pearson publications ,II Edition

3. Albert Lulushi, "Developing ORACLE FORMS Applications", Prentice Hall ,1997

Web resources: Web resources from NDL Library, E-content from open-source libraries

Course Code: 23U5CSDE02	Operating Systems		Credits: 3
Lecture Hours: (L) per week: 5	Tutorial Hours : (T) per week	Lab Practice Hours: (P)per week	Total: (L+T+P) per week: 5
Course Category :EC-5	Year & Semester:III Year V Semester		Admission Year:
Pre-requisite	Basic Knowledge on Computer and its functions		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

- Understanding the design of the Operating System
- Imparting knowledge on CPU scheduling, Process and Memory Management.
- To code specialized programs for managing overall resources and operations of the computer.

Course Outcomes: (for students: To know what they are going to learn)

CO1: Define the fundamentals of OS and identify the concepts relevant to process , process life cycle, Scheduling Algorithms, Deadlock and Memory management

CO2: know the critical analysis of process involving various algorithms, an exposure to threads and semaphores

CO3: Have a complete study about Deadlock and its impact over OS. Knowledge of handling Deadlock with respective algorithms and measures to retrieve from deadlock. .

CO4: Have complete knowledge of Scheduling Algorithms and its types.

CO5: understand memory organization and management

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	Contents	Required Hours
I	**Introduction**: operating system, history (1990s to 2000 and beyond), distributed computing, parallel computation. **Process concepts:** definition of process, process states-Life cycle of a process, process management- process state transitions, process control block(PCB), process operations , suspend and resume, context switching, Interrupts - Interrupt processing, interrupt classes, Inter process	09

	communication-signals, message passing.	
II	**Asynchronous concurrent processes:** mutual exclusion-critical section, mutual exclusion primitives, implementing mutual exclusion primitives, Peterson's algorithm,software solutions to the mutual Exclusion Problem-, n-thread mutual exclusion- Lamports Bakery Algorithm. Semaphores – Mutual exclusion with Semaphores, thread synchronization with semaphores, counting semaphores, implementing semaphores. **Concurrent programming:** monitors, message passing	**09**
III	**Deadlock and indefinite postponement:** Resource concepts, four necessary conditions for deadlock, deadlock prevention, deadlock avoidance and Dijkstra's Banker's algorithm, deadlock detection, deadlock recovery	**09**
IV	**Job and processor scheduling:** scheduling levels, scheduling objectives, scheduling criteria, preemptive vs non-preemptive scheduling, interval timer or interrupting clock, priorities, scheduling algorithms- FIFO scheduling, RR scheduling, quantum size, SJF scheduling, SRT scheduling, HRN scheduling, multilevel feedback queues, Fair share scheduling	**09**
V	**Real Memory organization and Management**:**:** Memory organization, Memory management, Memory hierarchy, Memory management strategies, contiguous vs non-contiguous memory allocation, single user contiguous memory allocation, fixed partition multiprogramming, variable partition multiprogramming, Memory swapping **Virtual Memory organization:** virtual memory basic concepts, multilevel storage organization, block mapping, paging basic concepts, segmentation, paging/segmentation systems. **Virtual Memory Management:** Demand Paging, Page replacement strategies	**09**

Learning Resources:

- **Recommended Texts**

1. H.M. Deitel, Operating Systems, Third Edition, Pearson Education Asia, 2011

- **Reference Books**

1. William Stallings, Operating System: Internals and Design Principles, Seventh Edition, Prentice-Hall of India, 2012.
2. A. Silberschatz, and P.B. Galvin., Operating Systems Concepts, Nineth Edition, John Wiley &Sons(ASIA) Pte Ltd.,2012

Web resources: Web resources from NDL Library, E-content from open-source libraries

Mapping with Programme Outcomes:

Map course outcomes for each course with programme outcomes (PO) in the 3-point scale of Strong, Medium and Low

CO/PSO	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5	PSO 6
CO 1	3	3	3	3	3	3
CO 2	3	3	3	3	3	3
CO 3	2	3	3	2	2	3
CO 4	3	3	3	2	3	2
CO 5	2	3	2	3	3	2
Weightage of course contributedto each PSO	13	15	14	13	14	13

Course Code: 23U5CSDE03	Data Mining and Warehousing		Credits:3
Lecture Hours: (L) per week: 4	**Tutorial Hours : (T) per week**	**Lab Practice Hours: (P)per week**	**Total: (L+T+P) per week: 4**
Course Category :	**Year & Semester: III Year V Semester**		**Admission Year: 2023-2024**
Pre-requisite	Basic concept of database knowledge		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

- To provide the knowledge on Data Mining and Warehousing concepts and techniques.
- To study the basic concepts of cluster analysis
- To study a set of typical clustering methodologies, algorithms, andapplications

Course Outcomes: (for students: To know what they are going to learn)

CO1: To understand the basic concepts and the functionality of the various data mining and data warehousing component

CO2: To know the concepts of Data mining system architectures

CO3: To analyse the principles of association rules

CO4: To get analytical idea on Classification and prediction methods.

CO5: To Gain knowledge on Cluster analysis and its methods.

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	Contents	Required Hours
I	Introduction: Data mining – Functionalities – Classification – Introduction to Data Warehousing – Data Preprocessing: Preprocessing the Data – Data cleaning – Data Integration and Transformation – Data Reduction	**09**
II	Data Mining, Primitives, Languages and System Architecture: Data Mining – Primitives – Data Mining Query Language, Architecture of Data mining Systems. Concept	**09**

	Description, Characterization and Comparison: Concept Description, Data Generalization and Summarization, Analytical Characterization, Mining Class Comparison – Statistical Measures	
III	Mining Association Rules: Basic Concepts – Single Dimensional Boolean Association Rules From Transaction Databases, Multilevel Association Rules from transaction databases – Multi dimension Association Rules from Relational Database and Data Warehouses	**09**
IV	Classification and Prediction: Introduction – Issues – Decision Tree Induction – Bayesian Classification – Classification of Back Propagation. Classification based on Concepts from Association Rule Mining – Other Methods. Prediction – Introduction – Classifier Accuracy.	**09**
V	Cluster Analysis: Introduction – Types of Data in Cluster Analysis, Petitioning Methods – Hierarchical Methods-Density Based Methods – GRID Based Method – Model based Clustering Method	**09**

Learning Resources:

- **Recommended Texts**

1. Han and M. Kamber, "Data Mining Concepts and Techniques", 2001, Harcourt India Pvt. Ltd, New Delhi.

- **Reference Books**

1.K.P. Soman, Shyam Diwakar, V. Ajay "Insight into Data Mining Theory and Practice ", Prentice Hall of India Pvt. Ltd, New Delhi

2.Parteek Bhatia, 'Data Mining and Data Warehousing: Principles and Practical Techniques', Cambridge University Press, 2019

Web resources: Web resources from NDL Library, E-content from open-source libraries

Mapping with Programme Outcomes:

Map course outcomes for each course with programme outcomes (PO) in the 3-point scale of Strong, Medium and Low

CO/PSO	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5	PSO 6
CO 1	3	3	3	3	3	3
CO 2	3	3	3	3	3	3
CO 3	2	2	3	3	2	3
CO 4	3	3	3	3	2	2
CO 5	3	3	3	3	3	3
Weightage of course contributedto each PSO	14	14	15	15	13	14

Third Year (Semester – VI)

Course Code: CC13	Computer Networks		Credits:4
Lecture Hours: (L) per week: 5	**Tutorial Hours : (T) per week**	**Lab Practice Hours: (P)per week**	**Total: (L+T+P) per week: 5**
Course Category :	**Year & Semester:III Year VI Semester**		**Admission Year:**
Pre-requisite	Basic knowledge on networking		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

- To understand the concept of Data communication and Computer network
- To get a knowledge on routing algorithms.
- To impart knowledge about networking and inter networking devices

To gain the knowledge on Security over Network communication

Course Outcomes: (for students: To know what they are going to learn)

CO1: To Understand the basics of Computer Network architecture, OSI and TCP/IP reference models

CO2: To gain knowledge on Telephone systems and Satellite communications

CO3: To impart the concept of Elementary data link protocols

CO4: To analyze the characteristics of Routing and Congestion control algorithms

CO5: To understand network security and define various protocols such as FTP, HTTP, Telnet, DNS

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	Contents	Required Hours
I	Introduction – Network Hardware – Software – Reference Models – OSI and TCP/IP Models – Example Networks: Internet, ATM, Ethernet and Wireless LANs - Physical Layer – Theoretical Basis for Data Communication - Guided Transmission Media	**17**
II	Wireless Transmission - Communication Satellites – Telephone System: Structure, Local Loop, Trunks and Multiplexing and Switching. Data Link Layer: Design Issues – Error Detection and Correction.	**17**
III	Elementary Data Link Protocols - Sliding Window Protocols – Data Link Layer in the Internet - Medium Access Layer – Channel Allocation Problem – Multiple Access Protocols –	**17**

	Bluetooth	
IV	Network Layer - Design Issues - Routing Algorithms - Congestion Control Algorithms – IP Protocol – IP Addresses – Internet Control Protocols.	**17**
V	Transport Layer - Services - Connection Management - Addressing, Establishing and Releasing a Connection – Simple Transport Protocol – Internet Transporet Protocols (ITP) - Network Security: Cryptography.	**17**

Learning Resources:

- **Recommended Texts**
  1. A. S. Tanenbaum, "Computer Networks", 4th Edition, Prentice-Hall of India, 2008.
- **Reference Books**
  1. B. A. Forouzan, "Data Communications and Networking", Tata McGraw Hill, 4th Edition, 2017.
  2. F. Halsall, "Data Communications, Computer Networks and Open Systems", Pearson Education, 2008.
  3. D. Bertsekas and R. Gallagher, "Data Networks", 2nd Edition, PHI, 2008.
  4. Lamarca, "Communication Networks", Tata McGraw- Hill, 2002

Web resources: Web resources from NDL Library, E-content from open-source libraries

Mapping with Programme Outcomes:

Map course outcomes for each course with programme outcomes (PO) in the 3-point scale of Strong, Medium and Low

CO/PSO	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5	PSO 6
CO 1	3	3	3	3	3	3
CO 2	3	3	3	2	3	3
CO 3	3	2	3	3	2	3
CO 4	3	2	3	3	3	3
CO 5	3	3	3	2	3	3
Weightage of course contributedto each PSO	15	13	15	13	14	15

Course Code: CC14	.Net Programming		Credits: 4
Lecture Hours: (L) per week: 5	Tutorial Hours : (T) per week	Lab Practice Hours: (P)per week	Total: (L+T+P) per week:5
Course Category :CC14	Year & Semester: III Year VI Semester		Admission Year:
Pre-requisite	Basic knowledge on web programming		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

1. To develop ASP.NET Web application using standard controls.
2. To create rich database applications using ADO.NET.
3. To implement file handling operations.
4. To utilize ASP.NET security features for authenticating the web site.
5. To handles SQL Server Database using ADO.NET.

Course Outcomes: (for students: To know what they are going to learn)

CO1: To identify and **understand** the goals and objectives of the .NET framework and ASP.NET with C# language.

CO2: To **develop** web application using various controls.

CO3: To analyze C# programming techniques in developing web applications.

CO4: To assess a Web application using Microsoft ADO.NET.

CO5: To **develop** a software to solve real-world problems using ASP.NET

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	Contents	Required Hours
I	Overview of .NET framework: Common Language Runtime (CLR), Framework Class Library- C# Fundamentals: Primitive types and Variables – Operators - Conditional statements -Looping statements – Creating and using Objects – Arrays – String operations.	17
II	Introduction to ASP.NET - IDE-Languages supported	17

	Components -Working with Web Forms – Web form standard controls: Properties and its events – HTML controls -List Controls: Properties and its events.	
III	Rich Controls: Properties and its events – validation controls: Properties and its events– File Stream classes - File Modes – File Share – Reading and Writing to files – Creating, Moving, Copying and Deleting files – File uploading.	**17**
IV	ADO.NET Overview – Database Connections – Commands – Data Reader - Data Adapter - Data Sets - Data Controls and its Properties - Data Binding	**17**
V	Grid View control: Deleting, editing, Sorting and Paging. XML classes – Web form to manipulate XML files - Website Security - Authentication - Authorization – Creating aWeb application.	**17**

Learning Resources:

- **Recommended Texts**

1. SvetlinNakov,VeselinKolev& Co, Fundamentals of Computer Programming with C#,Faber publication, 2019.
2. Mathew, Mac Donald, The Complete Reference ASP.NET, Tata McGraw-Hill ,2015.

- **Reference Books**

1. Herbert Schildt, The Complete Reference C#.NET, Tata McGraw-Hill,2017.
2. Kogent Learning Solutions, C# 2012 Programming Covers .NET 4.5 Black Book, Dreamtech pres,2013.
3. Anne Boehm, Joel Murach, Murach's C# 2015, Mike Murach& Associates Inc. 2016.
4. DenielleOtey, Michael Otey, ADO.NET: The Complete reference, McGraw Hill,2008.
5. Matthew MacDonald, Beginning ASP.NET 4 in C# 2010, APRESS,2010.

Web resources: Web resources from NDL Library, E-content from open-source libraries

Mapping with Programme Outcomes:

Map course outcomes for each course with programme outcomes (PO) in the 3-point scale of Strong, Medium and Low

CO/PSO	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5	PSO 6
CO 1	3	3	3	3	3	2
CO 2	3	3	3	3	3	3
CO 3	2	3	3	2	2	3
CO 4	3	3	3	3	3	2
CO 5	2	3	3	3	3	3
Weightage of course contributedto each PSO	13	15	15	14	14	13

<table>
<tr><td>Course Code: CC15</td><td colspan="2">.Net Programming Lab</td><td>Credits: 4</td></tr>
<tr><td>Lecture Hours: (L) per week</td><td>Tutorial Hours : (T) per week</td><td>Lab Practice Hours: (P)per week: 5</td><td>Total: (L+T+P) per week: 5</td></tr>
<tr><td>Course Category :CC14</td><td colspan="2">Year & Semester: III Year VI Semester</td><td>Admission Year:</td></tr>
<tr><td>Pre-requisite</td><td colspan="3">Basic knowledge on</td></tr>
<tr><td colspan="4">Learning Objectives: (for teachers: what they have to do in the class/lab/field)
1. To develop ASP.NET Web application using standard controls.
2. To create rich database applications using ADO.NET.
3. To implement file handling operations.
4. To utilize ASP.NET security features for authenticating the web site.
5. To handles SQL Server Database using ADO.NET.</td></tr>
<tr><td colspan="4">Course Outcomes: (for students: To know what they are going to learn)
CO1: To identify and understand the goals and objectives of the .NET framework and ASP.NET with C# language.
CO2:To develop web application using various controls.
CO3:To analyze C# programming techniques in developing web applications.
CO4: To assess a Web application using Microsoft ADO.NET.
CO5: To develop a software to solve real-world problems using ASP.NET</td></tr>
<tr><td colspan="4">Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)</td></tr>
<tr><td></td><td colspan="2">List of Exercises:</td><td>Required Hours</td></tr>
<tr><td></td><td colspan="2">1. Create an exposure of Web applications and tools
2. Implement the Html Controls
3. Implement the Server Controls
4. Web application using Web controls.
5. Web application using List controls.</td><td></td></tr>
</table>

	6. Web Page design using Rich control. Validate user input using Validation controls. Working with File concepts. 7. Web application using Data Controls. 8. Data binding with Web controls 9. Data binding with Data Controls. 10. Database application to perform insert, update and delete operations. 11. Database application using Data Controls to perform insert, delete, edit, paging and sorting operation. 12. Implement the Xml classes. 13. Implement Authentication – Authorization. 14. Ticket reservation using ASP.NET controls. Online examination using ASP.NET controls	

Learning Resources:

- **Recommended Texts**

1. SvetlinNakov,VeselinKolev& Co, Fundamentals of Computer Programming with C#,Faber publication, 2019.
2. Mathew, Mac Donald, The Complete Reference ASP.NET, Tata McGraw-Hill ,2015.

- **Reference Books**

1. Herbert Schildt, The Complete Reference C#.NET, Tata McGraw-Hill,2017.
2. Kogent Learning Solutions, C# 2012 Programming Covers .NET 4.5 Black Book, Dreamtech pres,2013.
3. Anne Boehm, Joel Murach, Murach's C# 2015, Mike Murach& Associates Inc. 2016.
4.DenielleOtey, Michael Otey, ADO.NET: The Complete reference, McGraw Hill,2008.
5.Matthew MacDonald, Beginning ASP.NET 4 in C# 2010, APRESS,2010.

Web resources: Web resources from NDL Library, E-content from open-source libraries

Course Code: EC7	Introduction to Data Science		Credits: 3
Lecture Hours: (L) per week:5	Tutorial Hours : (T) per week	Lab Practice Hours: (P)per week	Total: (L+T+P) per week:6
Course Category : EC7	Year & Semester: III Year VI Semester		Admission Year:
Pre-requisite	Basic knowledge on Data and statistics		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

- To introduce the concepts, techniques and tools in Data Science
- To understand the various facets of data science practice, including data collection and integration, exploratory data analysis, predictive modeling, descriptive modeling and effective communication.

Course Outcomes: (for students: To know what they are going to learn)

CO1: To describe what Data Science is, what Statistical Inference means, identify probability distributions, fit a model to data and use tools for basic analysis and communication

CO2: To describe what Data Science is, what Statistical Inference means, identify probability distributions, fit a model to data and use tools for basic analysis and communication

CO3: To describe what Data Science is, what Statistical Inference means, identify probability distributions, fit a model to data and use tools for basic analysis and communication

CO4: To describe what Data Science is, what Statistical Inference means, identify probability distributions, fit a model to data and use tools for basic analysis and communication

CO5: To describe what Data Science is, what Statistical Inference means, identify probability distributions, fit a model to data and use tools for basic analysis and communication

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	Contents	Required Hours
I	**Introduction:** Benefits and uses – Facets of data – Data science process – Big data ecosystem and data science	17
II	**The Data science process**: • Overview – research goals - retrieving data - transformation – Exploratory Data Analysis – Model	17

	building	
III	**Algorithms** : • Machine learning algorithms – Modeling process – Types – Supervised – Unsupervised - Semi-supervised	**17**
IV	**Introduction to Hadoop** : • Hadoop framework – Spark – replacing MapReduce– NoSQL – ACID – CAP – BASE – types	**17**
V	**Case Study**: • Prediction of Disease - Setting research goals - Data retrieval – preparation - exploration - Disease profiling - presentation and automation	**17**

Learning Resources:

- **Recommended Texts**

1. Davy Cielen, Arno D. B. Meysman, Mohamed Ali, "Introducing Data Science", manning publications 2016

- **Reference Books**

1. Roger Peng, "The Art of Data Science", lulu.com 2016.
2. MurtazaHaider, "Getting Started with Data Science – Making Sense of Data with Analytics", IBM press, E-book.
3. Davy Cielen, Arno D.B. Meysman, Mohamed Ali,"Introducing Data Science: Big Data, Machine Learning, and More, Using Python Tools", Dreamtech Press 2016.
4. Annalyn Ng, Kenneth Soo, "Numsense! Data Science for the Layman: No Math Added", 2017,1st Edition.
5. Cathy O'Neil, Rachel Schutt, "Doing Data Science Straight Talk from the Frontline", O'Reilly Media 2013.
6. Lillian Pierson, "Data Science for Dummies", 2017 II Edition

Web resources: Web resources from NDL Library, E-content from open-source libraries

Mapping with Programme Outcomes:

Map course outcomes for each course with programme outcomes (PO) in the 3-point scale of Strong, Medium and Low

CO/PSO	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5	PSO 6
CO 1	3	3	3	3	3	3
CO 2	3	3	3	3	3	3
CO 3	3	2	3	2	3	3
CO 4	2	3	3	3	3	3
CO 5	2	3	3	3	3	2
Weightage of course contributedto each PSO	13	14	15	14	15	14

Course Code: EC8	Cyber Security		Credits: 3
Lecture Hours: (L) per week	Tutorial Hours : (T) per week	Lab Practice Hours: (P)per week	Total: (L+T+P) per week: 5
Course Category :EC8	Year & Semester:III Year VI Semester		Admission Year:
Pre-requisite	Basic skills on internet and its functions		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

The students will be able to

- Understand various block cipher and stream cipher models
- Describe the principles of public key cryptosystems, hash functions and digital signature
- To get a firm knowledge on Cyber Security Essentials

Course Outcomes: (for students: To know what they are going to learn)

CO1: Implement basic security algorithms required by any computing system

CO2: Analyze the vulnerabilities in any computing system and hence be able to design a security solution

CO3: Analyze the possible security attacks in complex real time systems and their effective countermeasures

CO4: Differentiate various governing bodies of cyber laws

CO5: Impart various privacy policies for an organization

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	Contents	Required Hours
I	**Introduction to Security** Data Encryption Standard-Block cipher principles-block cipher modes of operation-Advanced Encryption Standard (AES)-Triple DES-Blowfish-RC5 algorithm.	12
II	**Public Key Cryptography and Hash Algorithms** Principles of public key cryptosystems-The RSA algorithm-Key management - Diffie Hellman Key exchange- Hash functions-Hash Algorithms (MD5, Secure Hash Algorithm	12
	.	

III	**Fundamentals of Cyber Security** How Hackers Cover Their Tracks- Fraud Techniques- Threat Infrastructure- Techniques to Gain a Foothold (Shellcode, SQL Injection, Malicious PDF Files)- Misdirection, Reconnaissance, and Disruption Methods	12
IV	**Planning for Cyber Security** Privacy Concepts -Privacy Principles and Policies - Authentication and Privacy - Data Mining - Privacy on the Web - Email Security - Privacy Impacts of Emerging Technologies.	12
V	**Cyber Security Management** Security Planning - Business Continuity Planning - Handling Incidents - Risk Analysis - Dealing with Disaster – Legal Issues – Protecting programs and Data – Information and the law – Rights of Employees and Employers - Emerging Technologies - The Internet of Things - Cyber Warfare.	12

Learning Resources:

- **Recommended Texts**

  1. William Stallings, "Cryptography and Network Security", Pearson Education, 6th Edition,2013.

  2. Charles P. Pfleeger Shari Lawrence Pfleeger Jonathan Margulies, Security in Computing, 5th Edition , Pearson Education , 2015.

- **Reference Books**

  1. Graham, J. Howard, R., Olson, R., Cyber Security Essentials, CRC Press, 2011.
  2. George K.Kostopoulous, Cyber Space and Cyber Security, CRC Press, 2013.

Web resources: Web resources from NDL Library, E-content from open-source libraries

Mapping with Programme Outcomes:

Map course outcomes for each course with programme outcomes (PO) in the 3-point scale of Strong, Medium and Low

CO/PSO	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5	PSO 6
CO 1	3	3	3	2	3	3
CO 2	3	3	3	3	3	3
CO 3	3	3	3	2	3	2

CO 4	2	3	3	3	3	3
CO 5	3	3	3	3	2	2
Weightage of course contributedto each PSO	14	15	15	13	14	13