SRI SARADA COLLEGE FOR WOMEN (AUTONOMOUS)

SALEM - 16.

Reaccredited with B++ Grade by NAAC

(Affiliated to Periyar University)

PG & RESEARCH DEPARTMENT OF COMPUTER SCIENCE

OUTCOME BASED SYLLABUS

B.Sc. Computer Science

(For the students admitted in 2023 - 24 onwards)

(I Semester & II Semester)

Programme Outcomes :

PO1 To apply knowledge of computing appropriate to the discipline

PO2 To identify, formulate, and develop solutions to computational challenges based on ethical principles.

PO3 To design, implement, and evaluate a computational system to meet desired needs within realistic constraints.

PO4 To equip students with sufficient knowledge in web based programming languages for research project management.

PO5 To use appropriate techniques, skills and tools necessary for sustainable development of societal and environmental contexts.

PO6 To apply programming skills with their enhanced creativity as an individual or team.

Programme Specific Outcomes

PSO1: Think in a critical and logical based manner

PSO2: Familiarize the students with suitable software tools of computer science and industrial applications to handle issues and solve problems in mathematics or statistics and real time application related sciences.

PSO3: Know when there is a need for information, to be able to identify, locate, evaluate, and effectively use that information for the issue or problem at hand.

PSO4: Understand, formulate, develop programming model with logical approaches to Address issues arising in social science, business and other contexts.

PSO5: Acquire good knowledge and understanding to solve specific theoretical and applied problems in advanced areas of Computer science and Industrial statistics.

PSO6: Provide students/learners sufficient knowledge and skills enabling them to undertake further studies in Computer Science or Applications or Information Technology and its allied areas on multiple disciplines linked with Computer Science

SRI SARADA COLLEGE FOR WOMEN (AUTONOMOUS), SALEM - 16.

PG & RESEARCH DEPARTMENT OF COMPUTER SCIENCE

B.Sc. Computer Science

PROGRAMME STRUCTURE UNDER CBCS

(For the students admitted in 2023-24 onwards)

Total Credits: 140 + Extra Credits (Maximum 28)

I SEMESTER

Part	Course	Course Title	Code	Hrs./ Week	Credits
I	Language	Tamil/Hindi/Sanskrit - I	23ULTC1/ 23ULHC1/ 23ULSC1	6	3
II	General English	English - I	23ULEC1	6	3
III	Core Course - I	Python Programming	23UCSCC1	5	5
III	Core Course - II	Python Programming - Practical	23UCSCCQ1	5	5
III	Elective - I (GE): Generic Course	Numerical Methods	23UCSGEC1	4	3
IV	Skill Enhancement Course - I :	NME : Office Automation	23UCSSECQ1	2	2
IV	Skill Enhancement (Foundation Course)	Problem Solving Techniques	23UCSSEFC	2	2
		Total		**30**	**23**
V	Articulation and Idea Fixation Skills				
	Physical Fitness Practice - 35 hours per Semester				
	Advanced Diploma in Computer Programming Level - 1 : Certificate Course - 100 hours per year				

II SEMESTER

Part	Course	Course Title	Code	Hrs./ Week	Credits
I	Language	Tamil/Hindi/Sanskrit - II	23ULTC2/ 23ULHC2/ 23ULSC2	6	3
II	General English	English - II	23ULEC2	6	3
III	Core Course - III	Data Structures and Algorithms	23UCSCC2	5	5
III	Core Course - IV	Data Structures and Algorithms - Practical	23UCSCCQ2	5	5
III	Elective - II (GE): Generic Course	Graph Theory and its Applications	23UCSGEC2	4	3
IV	Skill Enhancement Course - II	NME(IKS):Foundation of Computer Science with Ethics	23UCSSEC2	2	2
IV	Skill Enhancement Course - III	Cyber Security - Practical	23UCSSECQ3	2	2
		Total		**30**	**23**
VI	Articulation and Idea Fixation Skills - 1 Extra Credit				
	Physical Fitness Practice - 35 hours per Semester - 1 Extra Credit				
	Advanced Diploma in Computer Programming Level - 1 : Certificate Course - 100 hours per year - 2 Extra Credits				
	Extra credits are given for extra skills and courses qualified in MOOC/NPTEL				

Subject Code	Subject Name	Category	L	T	P	S	Credits	Marks		
								CIA	External	Total
23UCSCC1	**Python Programming**	**Core**	**5**	**-**	**-**	**-**	**5**	**30**	**70**	**100**

Learning Objectives		
LO1	To make students understand the concepts of Python programming.	
LO2	To apply the OOPs concept in PYTHON programming.	
LO3	To impart knowledge on demand and supply concepts	
LO4	To make the students learn best practices in PYTHON programming	
LO5	To know the costs and profit maximization	
UNIT	**Contents**	**No. of Hours**
I	**Basics of Python Programming:** History of Python-Features of Python-Literal-Constants-Variables - Identifiers–Keywords-Built-in Data Types-Output Statements – Input Statements-Comments – Indentation- Operators-Expressions-Type conversions. **Python Arrays:** Defining and Processing Arrays – Array methods.	**15**
II	**Control Statements:** Selection/Conditional Branching statements: if, if-else, nested if and if-elif-else statements. Iterative Statements: while loop, for loop, else suite in loop and nested loops. **Jump Statements:** break, continue and pass statements**.**	**15**
III	**Functions:** Function Definition – Function Call – Variable Scope and its Lifetime-Return Statement. **Function Arguments**: Required Arguments, Keyword Arguments, Default Arguments and Variable Length Arguments- Recursion. **Python Strings:** String operations-Immutable Strings - Built-in String Methods and Functions - String Comparison. **Modules**: import statement- The Python module – dir() function – Modules and Namespace – Defining our own modules.	**15**
IV	**Lists:** Creating a list -Access values in List-Updating values in Lists-Nested lists -Basic list operations-List Methods. Tuples: Creating, Accessing, Updating and Deleting Elements in a tuple – Nested tuples– Difference between lists and tuples. **Dictionaries:** Creating, Accessing, Updating and Deleting Elements in a Dictionary – Dictionary Functions and Methods - Difference between Lists and Dictionaries.	**15**
V	**Python File Handling:** Types of files in Python - Opening and Closing files-Reading and Writing files: write() and writelines() methods-append() method – read() and readlines() methods – with keyword – Splitting words – File methods - File Positions- Renaming and deleting files.	**15**
	TOTAL HOURS	**75**

Course Outcomes		Programme Outcomes
CO	On completion of this course, students will	
CO1	Learn the basics of python, Do simple programs on python, Learn how to use an array.	PO1, PO2, PO3, PO4, PO5, PO6
CO2	Develop program using selection statement, Work with Looping and jump statements, Do programs on Loops and jump statements.	PO1, PO2, PO3, PO4, PO5, PO6
CO3	Concept of function, function arguments, Implementing the concept strings in various application, Significance of Modules, Work with functions, Strings and modules.	PO1, PO2, PO3, PO4, PO5, PO6
CO4	Work with List, tuples and dictionary, Write program using list, tuples and dictionary.	PO1, PO2, PO3, PO4, PO5, PO6
CO5	Usage of File handlings in python, Concept of reading and writing files, Do programs using files.	PO1, PO2, PO3, PO4, PO5, PO6
Textbooks		
1	ReemaThareja, "Python Programming using problem solving approach", First Edition, 2017, Oxford University Press.	
2	Dr. R. NageswaraRao, "Core Python Programming", First Edition, 2017, Dream tech Publishers.	
Reference Books		
1.	VamsiKurama, "Python Programming: A Modern Approach", Pearson Education.	
2.	Mark Lutz, "Learning Python", Orielly.	
3.	Adam Stewarts, "Python Programming", Online.	
4.	Fabio Nelli, "Python Data Analytics", APress.	
5.	Kenneth A. Lambert, "Fundamentals of Python – First Programs", CENGAGE Publication.	
Web Resources		
1.	https://www.programiz.com/python-programming	
2.	https://www.guru99.com/python-tutorials.html	
3.	https://www.w3schools.com/python/python_intro.asp	
4.	https://www.geeksforgeeks.org/python-programming-language/	
5.	https://en.wikipedia.org/wiki/Python_(programming_language)	

Mapping with Programme Outcomes:

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	2	3
CO 3	3	3	3	3	2	2
CO 4	3	3	3	3	2	3
CO 5	3	2	3	3	3	3

Weightage of course contributed to each PSO	15	14	15	15	13	14

S-Strong-3 M-Medium-2 L-Low-1

Subject Code	Subject Name	Category	L	T	P	S	Credits	Marks		
								CIA	External	Total
23UCSCCQ1	**Python Programming-Practical**	**Core**	-	-	5	-	5	30	70	100

Learning Objectives	
LO1	Be able to design and program Python applications.
LO2	Be able to create loops and decision statements in Python.
LO3	Be able to work with functions and pass arguments in Python.
LO4	Be able to build and package Python modules for reusability.
LO5	Be able to read and write files in Python.

LAB EXERCISES	**Required Hours**
1. Program using variables, constants, I/O statements in Python. 2. Program using Operators in Python. 3. Program using Conditional Statements. 4. Program using Loops. 5. Program using Jump Statements. 6. Program using Functions. 7. Program using Recursion. 8. Program using Arrays. 9. Program using Strings. 10. Program using Modules. 11. Program using Lists. 12. Program using Tuples. 13. Program using Dictionaries. 14. Program for File Handling.	**75**

Course Outcomes	
On completion of this course, students will	
CO1	Demonstrate the understanding of syntax and semantics of PYTHON language
CO2	Identify the problem and solve using PYTHON programming techniques.
CO3	Identify suitable programming constructs for problem solving.
CO4	Analyze various concepts of PYTHON language to solve the problem in an efficient way.
CO5	Develop a PYTHON program for a given problem and test for its correctness.

Mapping with Programme Specific Outcomes:

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

S-Strong-3 M-Medium-2 L-Low-1

<table>
<tr><td colspan="2">Title of the Course</td><td colspan="8">Numerical Methods</td></tr>
<tr><td colspan="2">Paper Number</td><td colspan="8">EC I (GENERIC COURSE)</td></tr>
<tr><td rowspan="2">Category</td><td rowspan="2">ELECTIVE COURSE</td><td>Year</td><td>I</td><td rowspan="2">Credits</td><td rowspan="2">3</td><td rowspan="2">Course Code</td><td rowspan="2">23UCSGEC1</td></tr>
<tr><td>Semester</td><td>I</td></tr>
<tr><td colspan="2" rowspan="2">Instructional Hours per week</td><td>Lecture</td><td colspan="2">Tutorial</td><td colspan="2">Lab Practice</td><td colspan="2">Total</td></tr>
<tr><td>4</td><td colspan="2">-</td><td colspan="2">-</td><td colspan="2">4</td></tr>
<tr><td colspan="2">Pre-requisite</td><td colspan="8">12th Standard Mathematics</td></tr>
<tr><td colspan="2">Objectives of the Course</td><td colspan="8">1. To introduce the various topics in Numerical methods.
2. To make understand the fundamentals of algebraic equations.
3. To apply interpolation and approximation on examples.
4. To solve problems using numerical differentiation and integration.
5. To solve linear systems, numerical solution of ordinary differential equations.</td></tr>
<tr><td colspan="10">Course Outcomes:
Students will be able to
CO1:Know how to solve various problems on numerical methods
CO2: Use approximation to solve problems
CO3: Differentiation and integration concept are applied
CO4: Apply, direct methods for solving linear systems
CO5: Find numerical solution of ordinary differential equations</td></tr>
<tr><td>Course Outline</td><td colspan="9">Unit–I(Hours: 15)
Fundamentals of Algebraic Equation
Solution of algebraic and transcendental equations-Bisection method – Fixed point iteration method – Newton Raphson method –linear system of equations – Gauss elimination method – Gauss Jordan method.
Chapter 3 (Sections 3.1, 3.2 & 3.4) & Chapter 4 (Sections 4.2 & 4.2.1)</td></tr>
<tr><td></td><td colspan="9">Unit –II(Hours: 15)
Iterative, Interpolation and Approximation
Iterative methods - Gauss Jacobi and Gauss Seidel – Eigen values of a matrix by Power method and Jacobi’s method for symmetric matrices. Interpolation with unequal intervals – Lagrange’s interpolation – Newton’s divided difference interpolation.
Chapter 4 (Sections 4.7 - 4.9), Chapter 13 (Section 13.1,13.2) & Chapter 8 (Sections 8.1-8.4, 8.5, 8.7)</td></tr>
<tr><td></td><td colspan="9">Unit–III(Hours: 15)
Interpolation with Equal Interval
Difference operators and relations. -Interpolation with equal intervals – Newton’s forward and backward difference formulae.
Chapter 5 (Section 5.1, 5.2) & Chapter 6 (Sections 6.1 - 6.3)</td></tr>
<tr><td></td><td colspan="9">Unit–IV(Hours: 15)
Numerical Differentiation and Integration
Approximation of derivatives using interpolation polynomials – Numerical integration using Trapezoidal, Simpson’s 1/3 rule.</td></tr>
</table>

	Chapter 9 (Sections 9.1- 9.4, 9.9 - 9.11 & 9.13)
	Unit –V (Hours:15) **Initial Value Problems for Ordinary Differential Equations** Single step methods – Taylor's series method – Euler's method – Modified Euler's method – Runge Kutta method for solving(first, second, Third and 4th) order equations – Multi step methods **Chapter 11 (Sections 11.5, 11.9, 11.11 - 11.13 & 11.16 - 11.18)**
Skills acquired from the course	Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication and Transferrable Skill
Recommended Text	P.Kandasamy, K. Thilagavathy, K.Gunavathy- Numerical Methods,First edition, S.Chand&CompanyLtd.
Reference Books	H.C.Saxena-FiniteDifferencesandNumericalAnalysis,S.ChandPublishers,2005.
Web resources	https://nptel.ac.in/

Subject Code	**Subject Name**	**Category**	**L**	**T**	**P**	**S**	**Credits**	**Inst. Hours**	**Marks**		
									CIA	**External**	**Total**
23UCSSECQ1	**Office Automation**	Skill Enhancement Course :NME	2	-	-	-	2	2	30	70	100

Learning Objectives	
LO1	Understand the basics of computer systems and its components.
LO2	Understand and apply the basic concepts of a word processing package.
LO3	Understand and apply the basic concepts of electronic spreadsheet software.
LO4	Understand and apply the basic concepts of database management system.
LO5	Understand and create a presentation using PowerPoint tool.

UNIT	**Contents**	**No. of Hours**
I	**Word Processing:** Open, Save and close word document; Editing text - tools, formatting, bullets ; Spell Checker - Document formatting - Paragraph alignment, indentation, headers and footers, numbering; printing - Preview, options, merge.	6
II	**Spreadsheets:** Excel opening, entering text and data, formatting, navigating; Formulas – entering, handling and copying; Charts - creating, formatting and printing, analysis tables, preparation of financial statements, introduction to data analytics.	6
III	**Database Concepts:** The concept of data base management system; Data field, records, and files, Sorting and indexing data; Searching records. Designing queries, and reports; Linking of data files; Understanding Programming environment in DBMS; Developing menu drive applications in query language(MS-Access).	6
IV	**Power point:** Introduction to Power point - Features –	6

	Understanding slide typecasting & viewing slides – creating slide shows. Applying special object – including objects & pictures – Slide transition – Animation effects, audio inclusion, timers.	
V	**Set-Up MS Teams** Chat on MS Teams - Different features of MS Teams - Calendar - Schedule a call on MS Teams - Scheduling Assistant - Out of Office- Teams - How to setup Teams - Make multiple channels on Teams- Approvals - Using approvals on MS Teams- Uploading files and folders - Sharing Access on One Drive - Different Sharing Access - Password protect for sharing purpose - Creating Shared Library - Creating Shared Library - Recycle Bin - Recycle Bin - Introduction to SharePoint - Introduction to SharePoint - Create Site - Create Site - Different features of SharePoint - Different features of SharePoint	6
	Total	**30**

Course Outcomes		Programme Outcomes
CO	On completion of this course, students will	
CO1	Possess the knowledge on the basics of computers and its components	PO1, PO2, PO3, PO6, PO8
CO2	Gain knowledge on Creating Documents, spreadsheet and presentation.	PO1, PO2, PO3, PO6
CO3	Learn the concepts of Database and implement the Query in Database.	PO3, PO5, PO7
CO4	Demonstrate the understanding of different automation tools.	PO3,PO4, PO5, PO7
CO5	Utilize the automation tools for documentation, calculation and presentation purpose.	PO4, PO6, PO7, PO8
Text Book		
1	PeterNorton,"IntroductiontoComputers"–TataMcGraw-Hill.	
Reference Books		
1.	Jennifer Ackerman Kettel, Guy Hat-Davis, Curt Simmons, "Microsoft 2003", Tata McGrawHill.	
Web Resources		
1.	https://www.udemy.com/course/office-automation-certificate-course/	
2.	https://www.javatpoint.com/automation-tools	

Mapping with Programme Specific Outcomes:

MAPPING TABLE						
CO/ PSO	**PSO 1**	**PSO 2**	**PSO 3**	**PSO 4**	**PSO 5**	**PSO 6**
CO1	3	2	1	2	2	2
CO2	2	3	1	3	2	2
CO3	1	3	1	1	3	1
CO4	1	2	1	1	3	1
CO5	1	2	1	1	3	3
Weightage of course contributed to each PSO	**8**	**12**	**5**	**8**	**13**	**9**

S - Strong - 3 **M - Medium - 2** **L - Low - 1**

Subject Code	Subject Name	Category	L	T	P	S	Credits	Inst. Hours	Marks		
									CIA	External	Total
23UCSSEFC	Problem Solving Techniques	Skill Enhancement (Foundation Course)	2	-	-	-	2	2	30	70	100

Learning Objectives	
LO1	Familiarize with writing of algorithms, fundamentals of C and philosophy of problem solving.
LO2	Implement different programming constructs and decomposition of problems into functions.
LO3	Use data flow diagram, Pseudo code to implement solutions.
LO4	Define and use of arrays with simple applications
LO5	Understand about operating system and their uses

UNIT	Contents	No. of. Hours
I	**Introduction:** History, characteristics and limitations of Computer. Hardware/Anatomy of Computer: CPU, Memory, Secondary storage devices, Input Devices and Output devices. Types of Computers: PC, Workstation, Minicomputer, Main frame and Supercomputer. Software: System software and Application software. **Programming Languages:** Machine language, Assembly language, High-level language,4 GL and 5GL-Features of good programming language. Translators: Interpreters and Compilers.	6
II	**Data:** Data types, Input, Processing of data, Arithmetic Operators, Hierarchy of operations and Output. Different phases in Program Development Cycle (PDC).**Structured Programming: Algorithm:** Features of good algorithm, Benefits and drawbacks of algorithm. **Flowcharts:** Advantages and limitations of flowcharts, when to use flowcharts, flowchart symbols and types of flowcharts.**Pseudocode:** Writing a pseudocode. Coding, documenting and testing a program: Comment lines and types of errors. **Program design:** Modular Programming.	6
III	**Selection Structures:** Relational and Logical Operators -Selecting from Several Alternatives – Applications of Selection Structures. **Repetition Structures:** Counter Controlled Loops –Nested Loops– Applications of Repetition Structures.	6
IV	**Data:** Numeric Data and Character Based Data. **Arrays:** One Dimensional Array - Two Dimensional Arrays – Strings as Arrays of Characters.	6
V	**Data Flow Diagrams:** Definition, DFD symbols and types of DFDs. **Program Modules:** Subprograms-Value and Reference parameters- Scope of a variable - Functions – Recursion. **Files:** File Basics-Creating and reading a sequential file- Modifying Sequential Files.	6
	TOTAL HOURS	**30**

Course Outcomes		Programme Outcomes
CO	On completion of this course, students will	
CO1	Study the basic knowledge of Computers.Analyze the programming languages.	PO1, PO2, PO3, PO4, PO5, PO6
CO2	Study the data types and arithmetic operations. Know about the algorithms. Develop program using flow chart and pseudocode.	PO1, PO2, PO3, PO4, PO5, PO6
CO3	Determine the various operators. Explain about the structures. Illustrate the concept of Loops	PO1, PO2, PO3, PO4, PO5, PO6
CO4	Study about Numeric data and character-based data. Analyze about Arrays.	PO1, PO2, PO3, PO4, PO5, PO6
CO5	Explain about DFD Illustrate program modules. Creating and reading Files	PO1, PO2, PO3, PO4, PO5, PO6
Textbooks		
1	**Stewart Venit,** "Introduction to Programming: Concepts and Design", Fourth Edition, 2010, Dream Tech Publishers.	
Web Resources		
1.	https://www.codesansar.com/computer-basics/problem-solving-using-computer.htm	
2.	http://www.nptel.iitm.ac.in/video.php?subjectId=106102067	
3.	http://utubersity.com/?page_id=876	

Mapping with Programme Specific Outcomes:

CO/PSO	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO 1	3	3	3	3	3	3
CO 2	3	3	3	3	3	3
CO 3	3	2	3	3	3	3
CO 4	3	3	2	3	3	3
CO 5	3	3	3	3	3	2
Weightage of course contributed to each PSO	**15**	**14**	**14**	**15**	**15**	**14**

S - Strong - 3 **M - Medium - 2** **L - Low - 1**

Title of the Course/ Paper	Subject Name	Category	L	T	P	S	Credits	Inst. Hours	Marks		
									CIA	External	Total
23UCSCC2	Data Structures and Algorithms	Core Course III	5	-	-	-	5	5	30	70	100

Learning Objectives		
LO1	To understand the concepts of ADTs	
LO2	To learn linear data structures-lists, stacks, queues	
LO3	To learn Tree structures and application of trees	
LO4	To learn graph structures and application of graphs	
LO5	To understand `various sorting and searching	
UNIT	**Contents**	**No. of Hours**
I	Abstract Data Types (ADTs)- List ADT-array-based implementation-linked list implementation singly linked lists-circular linked lists-doubly-linked lists-applications of lists-Polynomial Manipulation- All operations-Insertion-Deletion-Merge-Traversal	15
II	Stack ADT-Operations- Applications- Evaluating arithmetic expressions – Conversion of infix to postfix expression-Queue ADT-Operations-Circular Queue- Priority Queue- dequeue applications of queues.	15
III	Tree ADT-tree traversals-Binary Tree ADT-expression trees-applications of trees-binary search tree ADT- Threaded Binary Trees-AVL Trees- B-Tree- B+ Tree – Heap-Applications of heap.	15
IV	Definition- Representation of Graph- Types of graph-Breadth first traversal – Depth first traversal-Topological sort- Bi-connectivity – Cut vertex- Euler circuits-Applications of graphs.	15
V	Searching- Linear search-Binary search-Sorting-Bubble sort-Selection sort-Insertion sort-Shell sort-Radix sort-Hashing-Hash functions-Separate chaining- Open Addressing-Rehashing Extendible Hashing	15
	Total	**75**

Course Outcomes		Programme Outcome
CO	**On completion of this course, students will**	
CO1	Understand the concept of Dynamic memory management, data types, algorithms, Big O notation	PO1, PO6
CO2	Understand basic data structures such as arrays, linked lists, stacks and queues	PO2
CO3	Describe the hash function and concepts of collision and its resolution methods	PO2,PO4
CO4	Solve problem involving graphs, trees and heaps	PO4,PO6
CO5	Apply Algorithm for solving problems like sorting, searching, insertion and deletion of data	PO5,PO6
Text Books		
1	Mark Allen Weiss, "Data Structures and Algorithm Analysis in C++", Pearson Education 2014, 4th Edition.	
2	ReemaThareja, "Data Structures Using C", Oxford Universities Press 2014, 2nd Edition	
Reference Books		
1.	Thomas H.Cormen, Chales E.Leiserson, RonaldL.Rivest, Clifford Stein, "Introduction to Algorithms", McGraw Hill 2009, 3rd Edition.	
2.	Aho, Hopcroft and Ullman, "Data Structures and Algorithms", Pearson Education 2003	
Web Resources		
1.	https://www.programiz.com/dsa	
2.	https://www.geeksforgeeks.org/learn-data-structures-and-algorithms-dsa-tutorial/	

Mapping with Programme Outcomes:

S - Strong - 3 M - Medium - 2 L – Low - 1

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	1	3	3	3
CO 3	3	3	3	2	3	2
CO 4	3	2	3	2	3	3
CO 5	3	3	3	3	3	3
Weightage of course contributed to each PSO	15	14	13	13	15	14

Title of the Course/ Paper	Subject Name	Category	L	T	P	S	Credits	Inst. Hours	Marks		
									CIA	External	Total
23UCSCCQ2	Data Structures and Algorithms - Practical	Core Course	-	-	5	-	5	5	40	60	100

Learning Objectives

LO1	To understand the concepts of ADTs
LO2	To learn linear data structures-lists, stacks, queues
LO3	To learn Tree structures and application of trees
LO4	To learn graph structures and application of graphs
LO5	To understand various sorting and searching

Sl. No	Contents	No. of Hours
1.	Write a program to implement the List ADT using arrays and linked lists.	75
2.	Write programs to implement the following using a singly linked list. • Stack ADT • Queue ADT	
3.	Write a program that reads an infix expression, converts the expression to postfix form and then evaluates the postfix expression (use stack ADT).	
4.	Write a program to implement priority queue ADT.	
5.	Write a program to perform the following operations: • Insert an element into a binary search tree. • Delete an element from a binary search tree. • Search for a key element in a binary search tree.	
6.	Write a program to perform the following operations • Insertion into an AVL-tree • Deletion from an AVL-tree	
7.	Write programs for the implementation of BFS and DFS for a given graph.	
8.	Write programs for implementing the following searching methods: • Linear search • Binary search.	
9.	Write programs for implementing the following sorting methods: • Bubble sort • Selection sort • Insertion sort • Radix sort.	

	Total	75
Course Outcomes		**Programmem Outcome**
CO	**On completion of this course, students will**	
1	Understand the concept of Dynamic memory management, data types, algorithms, Big O notation	PO1,PO4,PO5
2	Understand basic data structures such as arrays, linked lists, stacks and queues	PO1, PO4,PO6
3	Describe the hash function and concepts of collision and its resolution methods	PO1,PO3,PO6
4	Solve problem involving graphs, trees and heaps	PO3,PO4
5	Apply Algorithm for solving problems like sorting, searching, insertion and deletion of data	PO1,PO5,PO6
Text Books		
1	Mark Allen Weiss, "Data Structures and Algorithm Analysis in C++", Pearson Education 2014, 4th Edition.	
2	ReemaThareja, "Data Structures Using C", Oxford Universities Press 2014, 2nd Edition	
Reference Books		
1	Thomas H. Cormen, Chales E.Leiserson, RonaldL. Rivest, Clifford Stein, "Introduction to Algorithms", McGraw Hill 2009, 3rd Edition	
2.	Aho, Hopcroft and Ullman, "Data Structures and Algorithms", Pearson Education 2003	
Web Resources		
1.	https://www.programiz.com/dsa	
2.	https://www.geeksforgeeks.org/learn-data-structures-and-algorithms-dsa-tutorial/	

Mapping with Programme Outcomes:

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	1	3	2	3
CO 3	3	3	3	3	2	3
CO 4	3	3	3	3	2	3
CO 5	3	2	3	3	3	3
Weightage of course contributed to each PSO	15	15	13	15	13	15

S-Strong-3 **M-Medium-2** **L-Low-1**

Course Code: 23UCSGEC2	Graph Theory and its Applications		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 : Elective II : Generic Course	Year & Semester: I Year & II Semester	Admission Year: 2023-24	
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)

1. Definition of Graph, sub graph their representations, degree and algebraic operations.
2. Connected graphs, weighted graphs and shortest paths
3. Trees: Characterizations, spanning tree, minimum spanning trees
4. Eulerian and Hamiltonian graphs: Characterization, Necessary and sufficient conditions
5. Special classes of graphs: Bipartite graphs, line graphs, chordal graphs.

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

CO1: To Introduce the fundamental concepts in graph theory Graphs, subgraphs, walks, Euler graphs, Hamiltonian Paths Tree Properties , Hamiltonian paths and circuits

CO2: Understanding the concepts of Circuits, Cut set and its Properties, Network Flows, Isomorphism and Combinatorial and Planar Graphs.

CO3: Applying the concept of Colouring with Chromatic Number, Directed Graphs, Matching Covering Pattern and Euler Graphs

CO4: Analysing the Various Concepts of Representation of Graphs, Euler Paths Circuit, Kruskals and Prims Algorithms, Connected Components.

CO5: Implementation of an application using All Types of Graphs and evaluate the Applications with travelling sales person Problem, K colour Problem with n vertices in a Graph and Shortest Path finding Problem using Directed and Undirected 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	**INTRODUCTION**: Graph-mathematical definition- Introduction – sub graphs –Walks, paths, Circuits connectedness- Components- Euler Graphs- Hamiltonian paths and circuits-Trees- properties of Trees- Distance and centers in Tree- Rooted and Binary Trees	12
II	**CONNECTIVITY AND PLANARITY:** Introduction to circuits - cut set- properties of cut set- All cut sets –connectivity and separability – Network Flows - 1-Isomorphism - 2-Isomorphism- Combinatorial and Geometric graphs- Planar Graphs – Different	12

	representation of planar graph.	
III	**COLORING AND DIRECTED GRAPH:** Basics of Colouring &Chromatic number – Chromatic partitioning – Graph Colouring – four colour Problem Chromatic polynomial - Matching – Covering - Directed graphs - Types of Directed Graphs – Diagraphs and binary relations – Directed paths- Euler Graph.	**12**
IV	**MATRIX REPRESENTATION IN GRAPH:** Matrix representation of graphs, Sub graphs& Quotient Graphs, Transitive Closure digraph, Euler's Path & Circuit (only definitions and examples), spanning Trees of Connected Relations, Prim's Algorithm to construct Spanning Trees, Weighted Graphs, Minimal, Spanning Trees by Prim's Algorithm & Kruskal's Algorithm.	**12**
V	**APPLICATIONS OF GRAPH:** Traveling Sales Person Problem with Directed and Un directed Graph, - Graph with n vertices and k colours- Shortest path from one to many Cities with directed graph-Shortest Paths with Un directed Graphs-Connected Components.	**12**
Extended Professional Component (is a part ofinternal component only, Not to be included inthe External Examination question paper)	Questions related to the above topics, from variouscompetitive examinations UPSC / TRB / NET / UGC – CSIR / GATE / TNPSC / others to be solved(To be discussed during the Tutorial hour)	
Skills acquired from the course	Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication andTransferrable Skill	

Learning Resources:

1 Narsingh Deo , " Graph Theory with Application to Engineering and Computer Science" Prentice Hall of India 2010(Reprint)

2 Rosen H "Discrete Mathematics and Its Application " Mc Graw Hill , 2007

Reference Books:

1 Discrete Maths for Computer Scientists & Mathematicians by Mott, Kandel, Baker

2 Clark J and Holton DA " First look at Graph Theory" Allied Publishers 1995

3 Discrete Maths for Computer Scientists & Mathematicians by Mott, Kandel, Baker

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

https://d3gt.com/

https://www.coursera.org/courses?query=graph%20theory

Course Code: 23UCSSEC2	Foundation of Computer Science with Ethics		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 : Skill Enhancement Course - II : NME (IKS)	Year & Semester: I Year II Semester		Admission Year: 2023-24
Pre-requisite	None		

Learning Objectives:

- To introduce students to the fundamental concepts and significance of computer science.
- To develop students' algorithmic thinking and problem-solving skills.
- To introduce students to the Indian Knowledge System and its relevance to computer science.
- To instill ethical considerations in computer science and emphasize the importance of responsible technology development.

Unit - I : Indian Contributions to Algorithmic Thinking

Exploration of ancient Indian mathematical and computational contributions, such as the development of algorithms for numerical calculations found in texts like the Sulba Sutras.- Relationship of early algorithms to modern algorithmic thinking in computer science.

Unit - II : Indian Philosophy and Ethics in Computing

Focus on Indian philosophical traditions, like Dharma and Karma - Application of ethical considerations in computer science - philosophies - responsibility and ethical decision-making in technology development.

Unit - III : Sanskrit and Natural Language Processing

Structured nature of the Sanskrit language and its relevance to natural language processing in computer science - Influence of Sanskrit grammar and linguistics in the development of language processing algorithms.

Unit - IV : Ancient Indian Architecture and Computer Systems Design

The relationship of architectural principles found in ancient Indian temple design to modern computer systems design - concepts of symmetry, modularity, and scalability.

Unit - V : Indian Traditional Knowledge and Sustainability in Computing

Relationship between traditional Indian knowledge to sustainable living and ecology, and application of eco-friendly technology and sustainable computing practices.

Books for References:

1. Computing with Python: An Introduction to Python for Science & Engineering by Charles Severance.
2. Ethics in Computing: A Concise Module by Miguel R. Luévano
3. The Man Who Knew Infinity: A Life of the Genius Ramanujan by Robert Kanigel
4. Computational Approaches to Sanskrit: Natural Language Processing by Amba Kulkarni and Gerard Huet
5. Indian Mathematics: Engaging with the World from Ancient to Modern Times edited by George Gheverghese Joseph
6. Computational Sustainability by Carla P. Gomes, Adele E. Howe, and Diana Marculescu
7. Relevant research papers, case studies, and online resources.

Course Outcomes: (for students: To know what they are going to learn)	
CO1	Understand the historical and cultural context of Indian knowledge systems and their relevance to computer science.
CO2	Understand ethical principles and responsible practices in computer science
CO3	Understand algorithmic thinking and problem-solving
CO4	Understand System and its Holistic approach

Course Code: 23UCSSECQ3	Cyber Security - Practical		Credits: 2
Lecture Hours: (L) per week:	Tutorial Hours : (T) per week	Lab Practice 2 Hours: (P)per week	Total: (L+T+P) per week: 2
Course Category : Skill Enhancement Course - III	Year & Semester: I Year II Semester		Admission Year: 2023-24
Pre-requisite	Basic Computer Knowledge		

Learning Objectives:

1. Deliver the fundamental understanding of Cyber Security.
2. Familiarize basic methods in Cyber Security
3. Explain various Cyber Security applications in society
4. Identify the key issues in online modes and safety methods used.

1. Checklist for reporting cyber crime at Cyber crime Police Station.
2. Checklist for reporting cyber crime online. 3. Reporting phishing emails.
3. Demonstration of email phishing attack and preventive measures.
4. Basic checklist, privacy and security settings for popular Social media platforms.
5. Reporting and redressal mechanism for violations and misuse of Social media platforms.
6. Configuring security settings in Mobile Wallets and UPIs. 8. Checklist for secure net banking.
7. Setting, configuring and managing three password policy in the computer (BIOS, Administrator and Standard User).
8. Setting and configuring two factor authentication in the Mobile phone.
9. Security patch management and updates in Computer and Mobiles.
10. Managing Application permissions in Mobile phone.
11. Installation and configuration of computer Anti-virus.
12. Installation and configuration of Computer Host Firewall. 15. Wi-Fi security management in computer and mobile

Course Outcomes		Programme Outcome
CO	On completion of this course, students will	
1	Outline the concepts of Cyber security	PO1, PO2
2	Apply the skill to practice the Cyber security platforms	PO1, PO2, PO3, PO4, PO5,PO6
3	Analyse the extensive procedures for Cyber security	PO1, PO2, PO3, PO4, PO5
4	Predict the performance of real time applications in Cyber security	PO1, PO2, PO3, PO4, PO5, PO6

Mapping with Programme Outcomes:

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

S-Strong-3 M-Medium-2 L-Low-1