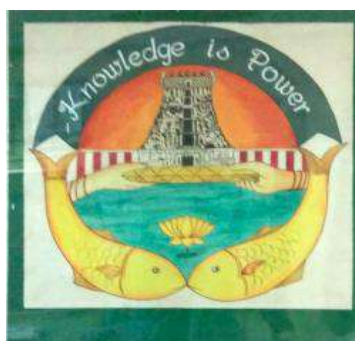


**SRI MEENAKSHI GOVT. ARTS COLLEGE FOR WOMEN
(AUTONOMOUS)**

MADURAI – 625 002.

PG AND RESEARCH DEPARTMENT OF COMPUTER SCIENCE



B.Sc. COMPUTER SCIENCE

SYLLABUS TO BE INTRODUCED FROM THE ACADEMIC

YEAR 2022 – 2025 (UNDER C.B.C.S)

SRI MEENAKSHI GOVERNMENT ARTS COLLEGE FOR WOMEN(A), MADURAI - 2

PG AND RESEARCH DEPARTMENT OF COMPUTER SCIENCE

BRIEF INTRODUCTION

The PG and Research Department of Computer Science was established in 1993 with B.Sc Computer Science. In the academic year 2007-2008 B.Sc Computer Science (Shift II) was started, and introduced M.Sc Computer Science in 2017, M. Phil Computer Science and Ph.D. Computer Science during the academic year 2018 - 2019. The sanctioned strength is 60 (25 + 25) for under graduate Programmes and 20 for post graduate Programme. Full time and Part time Research Programmes are offered and it was approved by Madurai Kamaraj University, Madurai. The Department is functioning successfully with Eight regular staff members and Four guest lecturers.

One well-equipped connected Computer lab is installed with a variety of software in the latest trends in computing such as Dot Net, Java based packages, Design packages, R Tool, Python, Networking, Windows Programming, Linux Programming, and other programming languages.

We focus on the areas for research such as Big Data, Cloud Computing, Grid Computing, Data Mining and Warehousing, Digital Image Processing, Artificial Intelligence, Wireless Networks, Algorithms etc, Our Faculty members acted as Resource Persons, Chairpersons and Presented papers in many International / National level Conferences, Workshops, and Seminars and also published research papers in various reputed Journals.

COURSES OFFERED

- **B. Sc. Computer Science (Shift I & Shift II)**
- **M. Sc Computer Science (Shift I)**
- **M. Phil. Computer Science (Full Time)**
- **Ph. D Computer Science (Part Time and Full Time)**

VISION

“TO STRIVE, TO SEEK AN TO SUCCEED”

Learning Outcome

- Apply a broad understanding of the fundamental theories, concepts, and applications of Computer Science in their career.
- Analyze a multifaceted computing problem and to apply principles of computing and other relevant disciplines to identify solutions and compare alternative solutions to computing problems.
- Apply Computer Science theory and software development fundamentals to produce computing-based solutions.
- To attain an ability to use current techniques, skills, and tools necessary for computing practice.
- To affiliate in a wide range of careers and/or graduate studies in computer science or related fields with a zeal for lifelong learning.
- To communicate effectively, both orally and in writing and engage in collaborative teamwork.
- Recognize the social and ethical errands of a professional working in the discipline.

MISSION

The mission of the department is to impart computer education to the students in the rural area of Madurai district, so that they become enlightened and intelligent, and to improve the standards of their life, as well as to produce graduates who excel in research and service. We also aim to inculcate the attitudes and values for empowerment of women that will motivate them towards the continuous process of learning and leadership. We strive to educate ground-breaking skills and technology for the benefit of learners through incessant upgradation of curriculum.

Programme Outcomes for Bachelor Programmes

PO 1: Computing Knowledge

To develop professionally competent citizens by applying the scientific knowledge of computer science with the ability to think clearly, rationally and creatively to support in evolving solutions to the social/public/scientific issues with responsible democratic participants.

PO 2: Critical Thinking

Ability and Willingness embark on new ventures, initiatives with critical thinking and desire for more continuous learning focusing on life skills.

PO 3: Design and Development of solutions

Design solutions for complex software problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal and environmental consideration.

PO 4: Communication

Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations and give and receive clear instructions.

PO 5: Individual and Team Work

Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

Programme Specific Outcomes for B.Sc Computer Science

PSO 1 :

To produce knowledgeable and skilled human resources who are employable in IT and ITES (Information Technology Enabled Services) by applying scientific knowledge of Computer Science with the ability to create and design innovative methodologies to solve complex problems for the betterment of the society.

PSO 2:

To develop academically competent and professionally motivated personnel, equipped with objective, critical thinking, right moral and ethical values that compassionately foster the scientific temper with a sense of social responsibility.

PSO 3:

To create solid knowledge in computer science, including software engineering, programming and languages, operating system, data structure and algorithms, databases, network, etc.

PSO 4:

Ability to produce an ethical software engineer and/or researcher in the evolving disciplines of computer science and engineering and its allied application domains by employing soft and project management skills learnt through internships, project work and/or collaborative projects with industry.

PSO 5:

Through Digital Literacy, understand, assess and commit to professional and ethical principles, norms and responsibilities of the cyber world and the ability for work efficacy as a part of a team and engage effectively with diverse stakeholders.

**SRI MEENAKSHI GOVT. ARTS COLLEGE FOR WOMEN (AUTONOMOUS),
MADURAI-2**

PG AND RESEARCH DEPARTMENT OF COMPUTER SCIENCE

B.Sc COMPUTER SCIENCE

TITLE OF THE PAPERS AND DISTRIBUTION OF CREDITS & MARKS

SEMESTER-I

Part	Course Type	Subject Code	Title of the Course	Hrs/Week	Credits	Exam Hrs	Marks		
							Int	Ext	Total
I	LC	U221A1/ U221H1	Tamil/Hindi	6	3	3	25	75	100
II	ELC	U222A1	English	6	3	3	25	75	100
III	CC -1	U22CS1	Programming in C	4	4	3	25	75	100
III	CC - 2	U22CS2P	Programming in C Lab	4	3	3	40	60	100
III	AC -1	U22ASS1	Allied – 1- Digital Electronics	4	4	3	25	75	100
III	CC - 3	U22CS3P	Office Automation Lab	4	3	3	40	60	100
IV	AEC -I	U22AE1	Value Education	2	2	3	25	75	100
Total				30	22				700

SEMESTER-II

Part	Course Type	Subject Code	Title of the Course	Hrs/Week	Credits	Exam Hrs	Marks		
							Int	Ext	Total
I	LC	U221A2/ U221H2	Tamil/Hindi	6	3	3	25	75	100
II	ELC	U222A2	English	6	3	3	25	75	100
III	CC - 4	U22CS4	Object Oriented Programming with C++	4	4	3	25	75	100
III	CC - 5	U22CS5P	Programming in C++ Lab	4	3	3	40	60	100
III	CC - 6	U22CS6	Data Structures and Algorithms	4	4	3	25	75	100
III	AC - 2	U22ASS2	Allied -2- Mathematical Foundations	4	4	3	25	75	100
IV	AEC - II	U22AE2	Environmental Studies	2	2	3	25	75	100
Total				30	23				700

SEMESTER-III

Part	Course Type	Subject Code	Title of the Course	Hrs/Week	Credits	Exam Hrs	Marks		
							Int	Ext	Total
I	LC	U221A3/ U221H3	Tamil/Hindi	6	3	3	25	75	100
II	ELC	U222A3	English	6	3	3	25	75	100
III	CC - 7	U22CS7	Computer System Architecture	4	4	3	25	75	100
III	CC - 8	U22CS8	Operating System	4	4	3	25	75	100
III	CC - 9	U22CS9P	Visual Basic Lab	4	3	3	40	60	100
III	AC - 3	U22ASS3	System Software	4	4	3	25	75	100
IV	NMEC-I	U22NMS1	NME I Information Technology	2	2	3	25	75	100
V			NCC/NSS/Extension Activity		1		100	-	100
Total				30	24				800

SEMESTER-IV

Part	Course Type	Subject Code	Title of the Course	Hrs/Week	Credits	Exam Hrs	Marks		
							Int	Ext	Total
I	LC	U221A4/ U221H4	Tamil/Hindi	6	3	3	25	75	100
II	ELC	U222A4	English	6	3	3	25	75	100
III	CC -10	U22CS10	Computer Networks	4	4	3	25	75	100
III	CC - 11	U22CS11	Database Management System	4	4	3	25	75	100
III	CC - 12	U22CS12P	ORACLE Lab	3	2	3	40	60	100
III	AC - 4	U22ASS4	Probability & Statistics	3	3	3	25	75	100
IV	NME C-II	U22NMS2	NME II Internet at your Finger Tips	2	2	3	25	75	100
IV	SEC- I	U22SES1	Skill Enhancement Course I Assembly Language Processor Lab	2	2	3	40	60	100
Total				30	23				800

SEMESTER-V

Part	Course Type	Subject Code	Title of the Course	Hrs/Week	Credits	Exam Hrs	Marks		
							Int	Ext	Total
III	CC – 13	U22CS13	Software Engineering	6	5	3	25	75	100
III	CC – 14	U22CS14	PHP and MySQL Programming	6	5	3	25	75	100
III	CC – 15	U22CS15P	PHP and MySQL Programming Lab	8	5	3	40	60	100
III	DSEC -I	U22DSS1A / U22DSS1B	Discipline Specific Elective Course-I 1.Computer Graphics 2. Artificial Intelligence	6	5	3	25	75	100
III	GEC I	U22GES1A/ U22GES1B	Generic Elective Course 1.Human Resource Management 2.Management Information System	2	2	3	25	75	100
IV	SEC– II	U22SES2	Skill Enhancement Course II Image Processing Tool - Lab	2	2	3	40	60	100
Total				30	24				600

SEMESTER-VI

Part	Course Type	Subject Code	Title of the Course	Hrs/Week	Credits	Exam Hrs	Marks		
							Int	Ext	Total
III	CC – 16	U22CS16	Advanced Java Programming	6	5	3	25	75	100
III	CC – 17	U22CSPW	Major Project	8	5	3	20	80	100
III	DSEC-II	U22DSS2A/ U22DSS2B	Discipline Specific Elective Course-II 1. Python Programming 2. Network Security	6	5	3	25	75	100
III	DSEC-III	U22DSS3AP/ U22DSS3BP	Discipline Specific Elective Course-III 1. Advanced Java Programming Lab 2. Network Lab and Android Programming	6	5	3	40	60	100
IV	SEC–III	U22SES3P	Skill Enhancement Course III Data Mining Tool - Lab	2	2	3	40	60	100
IV	AEC III	U22AE3	Ability Enhancement Course - III (General Knowledge)	2	2	3	25	75	100
Total				30	24				600

**COURSE STRUCTURE ABSTRACT FOR
B.Sc., PROGRAMME**

Part	Course	Total No of Papers	Hours	Credit	Marks	
I	Language Course (LC)	4	24	12	400	
II	English Language Course (ELC)	4	24	12	400	
III	Core Course (CC)	17	81	67	1700	
III	Allied Course (AC)	4	15	15	400	
III	Discipline Specific Elective Course (DSEC)	3	18	15	300	
III	Generic Elective Course (GEC)	1	2	2	100	
IV	Non Major Elective Course (NMEC)	2	4	4	200	
IV	Skill Enhancement Course (SEC)	3	6	6	300	
IV	Ability Enhancement Course(AEC)	Value Education	1	2	2	100
IV		Environmental Studies	1	2	2	100
IV		General Knowledge	1	2	2	100
V	NCC/NSS/Extension Activity	1	-	1	100	
Total		42	180	140	4200	
Value Added Course		2		4	100	
Total		44		144	4300	

Extra Credit Course

- The extra credit courses are optional one.
- The mode of evaluation is only by external examination for a maximum of 100Marks.

EXTRA CREDIT COURSES OFFERED

S.No. Exam	Code	Title of the Paper	Credits	Hours	Total Marks
1.	VAS1	Desk Top Publishing (DTP)	2	2	100
2.	VAS2	Python Programming	2	2	100

**SRI MEENAKSHI GOVT. ARTS COLLEGE FOR WOMEN
(A), MADURAI – 625 002.**

**PG AND RESEARCH DEPARTMENT OF COMPUTER SCIENCE
B. Sc COMPUTER SCIENCE**

Evaluation Pattern for UG Programme :

BLOOM'S TAXANOMY	INTERNAL	EXTERNAL
KNOWLEDGE	50%	50%
UNDERSTANDING	30%	30%
APPLY	20%	20%

Theory Paper:

Internal Evaluation

Test	20
Quiz/Assignment	5
Total	25

INTERNAL EVALUATION : 25
EXTERNAL EVALUATION : 75
TOTAL : 100

INTERNAL	:	NO MINIMUM
EXTERNAL	:	27 / 75
INTERNAL & EXTERNAL	:	40%

Question paper pattern for external Examination:

Section A	5/5 (2 marks each)	10
Section B	5/10 (Either or 5 marks each)	25
Section C	5/10 (Either or 8 marks each)	40
	Total	75

**SRI MEENAKSHI GOVT. ARTS COLLEGE FOR WOMEN
(AUTONOMOUS) MADURAI – 625 002.**

**PG AND RESEARCH DEPARTMENT OF COMPUTER SCIENCE B. Sc
COMPUTER SCIENCE**

Evaluation pattern for UG Programme:

Practical Paper:

Internal Evaluation:

Record	10
Viva-Voce	10
Internal Practical Exam	20
Total	40

INTERNAL EVALUATION	-	40
EXTERNAL EVALUATION	-	60
TOTAL	-	100

INTERNAL	:	NO
MINIMUM EXTERNAL	:	21 / 60
INTERNAL & EXTERNAL	:	40%

Evaluation pattern for UG Programme :

Project Paper:

Internal	20
External & Viva-Voce	80
Total	100

Programme : B. Sc Computer Science

Semester I

Sub. Code : U22CS1

Part III : Core

Hours : 4 P/W 60 HrsP/S

Credits : 4

TITLE OF THE PAPER: PROGRAMMING IN C

Pedagogy	Hours	Lecture	Peer Teaching	GD/VIDOES/TUTORIAL	ICT
	4	3	0 / 1	0 / 1	0 / 1
PREAMBLE: This course introduces the concept of programming in C Language.					
COURSE OUTCOME					Hrs P/S
At the end of the Semester, the Students will be able to					
UNIT 1	CO1: Define the basic concepts of C, operators and Expressions.				5
UNIT 2	CO2: Explain the role of managing input and output operations and Control statements				7
UNIT 3	CO3: Apply the working methodology of arrays and String functions.				15
UNIT 4	CO4: Demonstrate the concept of user defined functions, structure and union.				16
UNIT 5	CO5: Make use of pointers in c and its working principles.				17

Programme : B. Sc Computer Science

Part III : Core

Semester I

Hours : 4 P/W 60 HrsP/S

Sub. Code : U22CS1

Credits : 4

TITLE OF THE PAPER: PROGRAMMING IN C

UNIT I: OVERVIEW OF C: History of C – Importance of C – Sample C Programs – Basic Structure of C Program – Programming Style – Executing a C Program – CONSTANTS , VARIABLES AND DATA TYPES : Introduction – Character Set – C Tokens – Keywords and Identifiers – Constants – Variables – Data Types – Declaration of Variables – Declaration of Storage Class – Assigning Values to Variables – Defining Symbolic Constants. OPERATORS AND EXPRESSIONS: Introduction – Arithmetic Operators – Relational Operators – Logical Operators – Assignment Operators – Increment and Decrement Operators – Conditional Operator – Bitwise Operators – Special Operators – Arithmetic Expressions – Evaluation of Expressions – Precedence of Arithmetic Operators – Some Computational Problems – Type Conversion in Expressions – Operator Precedence and Associativity – Mathematical Functions.

UNIT II: MANAGING INPUT AND OUTPUT OPERATIONS – Introduction – Reading a Character – Writing a Character – Formatted Input – Formatted Output. DECISION MAKING AND BRANCHING: Introduction – Decision Making with IF Statement – Simple IF Statement – IF ELSE Statement – Nesting of IF ELSE Statement – ELSE IF Ladder – Switch Statement - Conditional Operator – GOTO Statement. DECISION MAKING AND LOOPING: Introduction – The WHILE Statement – The DO Statement – The FOR Statement – Jumps in loops.

UNIT III: ARRAYS: Introduction – One Dimensional Arrays – Two Dimensional Arrays – Initializing Two Dimensional Arrays – Multi Dimensional Arrays. HANDLING OF CHARACTER ARRAYS AND STRINGS: Introduction – Declaring and Initializing String Variables – Reading Strings From Terminal – Writing Strings To Screen – Arithmetic Operations On Characters – Putting String Together – Comparison of Two Strings String Handling Functions – Table of Strings.

UNIT IV: USER-DEFINED FUNCTIONS: Introduction – Need For User-Defined Functions – A Multi-Function Program – Element of User-Defined Functions – Definition of Functions – Return Values and Their Types – Function Calls – Function Declaration – Category of Functions – No Arguments and No Return Values – Arguments But No Return Values – Arguments with Return Values No Arguments But Returns a Value – Function that Return Multiple Values – Nesting of Functions. – Recursion – Passing Arrays to Functions – Passing Strings to Functions – The Scope, Visibility and Life Time of Variables. STRUCTURES AND UNIONS: Introduction – Defining a Structure – Declaring Structure Variables – Accessing Structure Members – Structure Initialization – Copying and Comparing Structure Variables

– Operations on Individual Members – Arrays of Structures – Arrays within Structures
– Structure within Structures – Structure and Functions – Unions – Size of Structures.

UNIT V: POINTERS: Introduction – Understanding Pointers – Accessing the Address of a Variables-Declaring Pointer Variables – Initialization of Pointer Variables – Accessing a Variable through its Pointers – Chain of Pointers – Pointer Expressions – Pointer Increments and Scale Factor – Pointer and Arrays – Pointers and Character Strings – Array of Pointers – Pointers as Function Arguments – Functions Returning Pointers – Pointers to Functions – Pointers and Structures. **FILE MANAGEMENT IN C:** Introduction – Defining and Opening a File – Closing a File – Input and Output Operations on Files.

TEXT BOOK:

Programming in ANSI C – by E. Balagurusamy, McGraw Hill - 6th Edition, 2012.

CHAPTERS:

UNIT – I : CHAPTERS: 1 – 1.1 - 1.10, 2 – 2.1 - 2.11, 3.

UNIT – II : CHAPTERS: 4, 5, 6 – 6.1 - 6.5.

UNIT – III : CHAPTERS: 7 – 7.1 - 7.7, 8 – 8.1 - 8.9.

UNIT – IV : CHAPTERS: 9 – 9.1 - 9.19, 10 – 10.1 - 10.13.

UNIT – V : CHAPTERS: 11 – 11.1- 11.16, 12 - 12.1 - 12.4.

REFERENCE BOOK:

1. Let us know C – Yeshwant Kanitkar.
2. The C Programming Language, B.W. Kernighan and D. M. Ritchie, ,2nd Edition PHI New Delhi,1990.

UNITS	TOPIC	LECTURE HOURS	MODE OF TEACHING
UNIT 1			
1.	Overview of C- importance ,Basic Structure, Data types.	1	Lecture
2.	Concept of C Tokens and Declaration of Storage Classes	1	Lecture
3.	Concept of types of Operators and Expressions	1	Lecture
4.	Type conversion in expressions and Operator Precedence and Associativity	1	Lecture
5.	Discussion	1	Group Discussion
UNIT 11			
6.	Managing Input / Output- Reading and Writing a character.	1	Lecture
7.	Concept of formatted input and output functions	1	Lecture
8.	Decision making and Branching- If statements-simple if, else if, nested if switch and GOTO statement.	1	Lecture
9.	Decision making and Looping – concepts of For statement, While and Do-While statement	1	Lecture
10.	Discussion	1	Group Discussion
11.	Implementation of Decision Making Branching and Looping	1	Peer Teaching
12.	Overview of Unit II	1	ICT (NPTEL Videos)
UNIT III			
13	Introduction about Array Concepts	1	Lecture
14.	Types of Arrays with examples	1	Lecture
15.	Introduction about Character Arrays	1	Lecture
16.	Exercise Problems	1	Tutorial
17.	Introduction about Strings	1	Lecture
18.	Reading a String from terminal and Writing a string to screen	1	Lecture
19.	Arithmetic operations on character	1	Lecture
20.	Putting string together without string handling function	1	Lecture
21.	Comparison of two strings without string functions	1	Lecture
22.	String Handling Functions	2	Lecture

23	Table of String	1	Lecture
24.	Exercise Problems of String Function	1	Tutorial
25	Array Implementation	1	Peer Teaching
26.	Array and String concepts	1	ICT (NPTEL Notes)
UNIT IV			
27	Introduction of user defined Function, Definition and Elements of User defined functions	1	Lecture
28.	User defined function return values and their types, Concept of Function Calls with example	1	Lecture
29.	Category of functions, Implementation of Function with multiple values return and nesting of functions	1	Lecture
30.	Recursive function	1	Lecture
31.	Passing arrays as argument to function, Passing strings to function, Scope and visibility and life time of variable	2	Lecture
32.	Implementation category of function	1	Tutorial
33.	Definition and Declaration structure variable	1	Lecture
34.	Structure initialization and Accessing structure members	1	Lecture
35.	Copying and comparing of structure variable	1	Lecture
36.	Array of structure and Array within structure	1	Lecture
37.	Structure within Structure, Structures and Functions	2	Lecture
38.	Concepts of Union	1	Lecture
39.	Exercise problems in structure	1	Tutorial
40.	Examples and implementation of Function	1	ICT (NPTEL Notes)
UNIT V			
41.	Introduction of pointers	1	Lecture
42.	Accessing the address of variables	1	Lecture
43.	Declaration and initialization of pointer variable	1	Lecture
44	Accessing variable through its pointer	1	Lecture
45.	Pointer expression with examples	1	Lecture
46.	Concepts of pointer and arrays	1	Lecture
47.	Concepts of pointers and character strings	2	Lecture
48.	Array of pointers	1	Lecture
49.	Exercise problems in pointers	1	Group Discussion

50.	Pointers in array examples	1	Peer Teaching
51.	Pointer as function argument and returning pointers	1	Lecture
52	Concept of pointers and structure	1	Lecture
53.	Introduction to file management in c	1	Lecture
54.	Definition of opening and closing a File	1	Lecture
55.	Concepts of input and output operations on files.	1	Lecture
56.	Overview of pointers and file	1	ICT (NPTEL Videos)

Course Outcomes (COs)	Programme Outcomes (POs)					Programme Specific Outcomes (PSOs)					Mean Scores of Cos
	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3	PSO4	PSO5	
CO1	2	3	3	3	3	3	2	4	3	4	3.00
CO2	3	3	4	4	3	4	3	3	3	3	3.30
CO3	4	3	3	3	4	4	3	4	3	3	3.40
CO4	4	3	3	3	3	4	3	3	4	3	3.30
CO5	3	3	3	3	4	3	3	3	4	3	3.20
Mean Overall Score											3.24

Result: The Score for this Course is 3.24 (High Relationship)

COURSE DESIGNER : Dr. A.S. BABY RANI
Associate Professor/ PG and Research Department of Computer Science

Programme : B. Sc Computer Science

Part III : Allied

Semester I

Hours : 4 P/W 60 Hrs P/S

Sub. Code : U22ASS1

Credits : 4

TITLE OF THE PAPER: DIGITAL ELECTRONICS

Pedagogy	Hours	Lecture	Peer Teaching	GD/VIDOES/TUTORIAL	ICT
	4	3	0 / 1	0 / 1	0 / 1
PREAMBLE: This course introduces the basic concepts of digital computer and electronic components.					
COURSE OUTCOME					Hrs P/S
At the end of the Semester, the Students will be able to					
UNIT 1	CO1: Classify the basics of Number System and Codes.				13
UNIT 2	CO2: Explain the concepts of Boolean Algebra and K-Maps.				15
UNIT 3	CO3: Identify the purpose and applications of Combinational and Sequential Logic				13
UNIT 4	CO4: Define the various Combinational and Sequential Logic.				9
UNIT 5	CO5: Outline about the Design of IoT, Domain specific IoT.				10

Programme : B. Sc Computer Science

Part III : Allied

Semester I

Hours : 4 P/W 60 Hrs P/S

Sub. Code : U22ASS1

Credits : 4

TITLE OF THE PAPER: DIGITAL ELECTRONICS

UNIT I NUMBER SYSTEM AND CODES

Number System and Codes: Why Binary Numbers Used – Binary-to-Decimal Conversion – Decimal- to-Binary Conversion – Octal Numbers – Hexadecimal Numbers – The ASCII Code – Excess-3 Code Gray Code – Arithmetic Circuits – Binary Addition – Binary Subtraction – Unsigned Binary Numbers – Sign Magnitude Numbers – 1's Complement – 2's Complement – 2's Complement Arithmetic – Arithmetic Building Block – half adder – full adder

UNIT II BOOLEAN ALGEBRA AND K-MAPS

Circuit Analysis and Design – Logic-Circuits – Inverter – OR gate – AND gate - Boolean Algebra - Boolean law & theorem – Sum of product – truth table to Karnaugh map – Pairs, Quads, and Octets – Karnaugh Simplification – Don't Care Conditions-Product of Sums Method – Product of Sum Simplification

UNIT III FLIP-FLOPS

Flip-flops – RS flip-flop – clocked RS flip-flop – D flip-flop – edge triggered D-Flip flop- JK flip flop- JK Master Slave flip-flop.

UNIT IV SHIFT REGISTER AND COUNTERS

Shift Registers – Types of Registers – Serial in Serial out – Serial in-Parallel Out – Parallel in Serial Out- Parallel in Parallel Out - Ring Counters – Counters – Asynchronous Counters (Ripple Counters), Mod-3 Counters, Mod-5 Counters.

UNIT V INTRODUCTION TO INTERNET OF THINGS

Introduction – Definition & Characteristics of IoT- Physical Design of IoT – Things in IoT- IoT Protocols. Domain Specific IoTs: Introduction – Home Automation – Cities – Environment – Energy – Retail – Logistics – Agriculture – Industry – Health & Life style.

TEXT BOOK (S):

1. Digital Principles & Applications – Malvino & Leech
2. Internet of Things, Arshdeep Bahga, Vijay Madisetti, Universities Press (INDIA) Private Ltd., 2015.

CHAPTERS:

UNIT I : CHAPTERS: 4 - 4.1 - 4.8, 5 - 5.1 - 5.7, 5.9
UNIT II : CHAPTERS: 1.2 - 1.7, 2 - 2.1 - 2.8
UNIT III : CHAPTERS: 8.1 - 8.4, 8.6 - 8.8,
UNIT IV : CHAPTERS: 10.1 - 10.6, 11.1, 11.4, 11.5
UNIT V : CHAPTERS: 1-1.1,1.2, 2-2.1-2.10

REFERENCE BOOK(S): Digital Logic and Computer Design, M.Morris Mano ,PHI, New Delhi 1994.

Programme : B. Sc Computer Science

Semester I

Sub. Code : U22ASS1

Part III : Allied

Hours : 4 P/W 60 Hrs P/S

Credits : 4

UNITS	TOPIC	LECTURE HOURS	MODE OF TEACHING
UNIT 1			
1.	Number System and Codes: Why Binary Numbers Used	1	Lecturer
2.	Binary-to-Decimal Conversion, Octal Numbers, Hexadecimal Numbers	1	Lecturer
3.	Exercise Problems	1	Group Discussion
4.	The ASCII Code	1	Lecturer
5.	Excess-3 Code	1	Lecturer
6.	Gray Code	1	Lecturer
7.	Arithmetic Circuits	1	Lecturer
8.	Binary Addition, Binary Subtraction, Unsigned Binary Numbers	1	Lecturer
9.	Sign Magnitude Numbers, 1's Complement, 2's Complement	1	Lecturer
10.	2's Complement Arithmetic	1	Lecturer
11.	Arithmetic Building Block	1	Lecturer
12.	half adder, full adder	1	Lecturer
13.	Example Problems	1	Tutorial
UNIT 11			
14.	Logic-Circuits – Inverter – OR gate – AND gate	1	Lecture
15.	Boolean Algebra	2	Lecture
16.	Boolean law & theorem	1	Lecture
17.	Example Problems	1	Tutorial
18.	Sum of product	2	Lecture
19.	Truth table to Karnaugh map – Pairs, Quads, and Octets	2	Lecture
20.	Karnaugh Simplification	2	Lecture
21.	Don't Care Conditions	1	Lecture
22.	Example Problems	1	Tutorial
23.	Product of Sums Method	1	Lecture
24.	Product of Sum Simplification	1	Lecture

UNIT III			
25.	Flip-flops	1	Lecture
26.	RS flip-flop	2	Lecture
27.	Clocked RS flip-flop	1	Lecture
28.	D flip-flop	2	Lecture
29.	Edge triggered D-Flip flop	1	Lecture
30.	JK flip flop	2	Lecture
31.	JK Master Slave flip-flop	2	Lecture
32.	Flip flops	2	Group Discussion
UNIT IV			
33.	Shift Registers -Types of registers: Serial in Serial out, Serial in-Parallel Out.	2	Lecture
34.	Parallel in Serial Out - Parallel in Parallel Out	1	Lecture
35.	Ring counters	1	Lecture
36.	Asynchronous Counters (Ripple Counters)	1	
37.	Mod-3-Counter, Mod-5-Counter	2	Lecture
38.	Counters	2	Group Discussion
UNIT V			
39.	Introduction: Definition & Characteristics of IoT	2	Lecture
40.	Physical Design of IoT- Things in IoT-IoT Protocols.	2	Lecture
41.	Domain Specific IoTs: Introduction – Home Automation , Cities	2	Lecture
42.	Environment, Energy, Retail, Logistics, Agriculture, Industry, Health & Life style.	2	Lecture
43.	Discussion	2	Group Discussion

Course Outcomes (COs)	Programme Outcomes (POs)					Programme Specific Outcomes (PSOs)					Mean Scores of Cos
	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3	PSO4	PSO5	
CO1	3	3	3	3	2	3	3	4	3	3	3.00
CO2	4	4	4	3	2	4	4	4	3	3	3.50
CO3	4	4	4	3	2	4	3	4	3	3	3.4
CO4	4	4	4	3	2	4	4	4	3	3	3.6
CO5	4	4	4	3	2	3	3	3	3	2	3.1
Mean Overall Score											3.32

Result: The Score for this Course is 3.32 (High Relationship)

COURSE DESIGNER : Dr. G. SUJATHA
Associate Professor / Department of Computer Science.

Programme : B. Sc Computer Science

Semester I

Sub. Code :U22CS2P

Part III : Core

Hours : 4 P/W 60 Hrs P/S

Credits : 3

TITLE OF THE PAPER: PROGRAMMING IN C LAB

1. Write a C program to check the given number is positive or negative.
2. Write a C program to find largest among three numbers using conditional operator.
3. Write a C program to calculate Electric Bill.
4. Write a C program to calculate Student Grade.
5. Write a C program to loan calculation using conditional statement.
6. Write a C program to calculate the value of $y=x^n$.
7. Write a C program to generate Multiplication Table.
8. Write a C program to sum of squares of n numbers.
9. Write a C program for addition of two Matrices.
10. Write a C program to Matrix Subtraction.
11. Write a C program to Matrix Multiplication.
12. Write a C program to Matrix Transpose.
13. Write a C program to get factorial of a given number.
14. Write a C program to find out largest and smallest element of an array.
15. Write a C program to check whether the given number is Prime or not.
16. Write a C program to create a function with no arguments and no return values.
17. Write a C program to create a function with arguments and no return values.
18. Write a C program to create a function with arguments and with return values.
19. Write a C program to create a function with no arguments but return values.
20. Write a C program to perform string copy in C without using STRCPY.
21. Write a C program for concatenation two strings without using STRCAT.
22. Write a C program to print the given names in alphabetical order.
23. Write a C program to perform string handling operations.
24. Write a C program to find Grade, Average and Result using structures.
25. Write a C program to sort the array in an Ascending order.
26. Write a C program to sort the array in Descending order.
27. Write a C program to Factorial Calculation using Recursion.

Programme : B. Sc Computer Science

Part III : SBE

Semester I

Hours : 4 P/W 60 Hrs P/S

Sub. Code : U22CS3P

Credits : 3

TITLE OF THE PAPER: OFFICE

AUTOMATION LABWORD

1. Open a Word document to prepare your Resume by performing the following operations.
 - (a) Formatting the Text- Alignment & Font style
 - (b) Page setup (margin alignment, page height & width)
2. Create a student mark sheet using table, find out the total & average marks and display the result.
3. Design an invitation of your course inauguration function using different fonts, font sizes, bullets and Word Art/ Clip Art
4. Mail Merge Concept
 - (a) Prepare an invitation and to be sent to specific addresses in the data source.

EXCEL

1. Create suitable work sheet with student mark details and use Data sort to display results and make out a suitable chart.
2. Prepare salary bill in a worksheet showing Basic Pay, DA, HRA, Gross salary, PF, Tax and Net salary using suitable Excel Functions.

POWER POINT

1. Create a power point presentation to explain various aspects of your college using auto play.
2. Create a power point presentation to explain the sales performance of a company over a period of five years. Include slides covering the profile of the company, year wise sales and graph with gridlines, legends and titles for axes. Use Clip Art and animation features.

ACCESS

1. Create a table for storing marks of 10 students. The fields of the table are given below: Reg. No., Name, Mark1, and Mark2, Mark3, Test average (Best Two /2), Assignment, Seminar and Total marks (Test average + Assignment + Seminar) The fields 'Mark1', 'Mark2', 'Mark3' should not allow the user to enter a mark greater than 25 and should display proper message in such case. Similar constraint for the field 'Assignment' is 5 marks and for the field 'Seminar', it is 10 marks.
2. Create a table showing names of authors of at least 10 different books, title of books, the prices of these books, name of publishers and year of publication. Also create Select, Action and Cross-tab queries to display the records from this table meeting the criteria used in these queries.
 1. Create a form to enter the data directly into this form. The fields required are: Basic Pay, DA, HRA, Gross salary, PF, Income tax and Net salary.
 2. Create a report that displays the customer name, address, phone number, Item code, product quantity of the customers whose orders have been pending for over a month.

Programme : B. Sc Computer Science

Semester II

Sub. Code : U22CS4

Part III : Core

Hours : 4 P/W 60 Hrs P/S

Credits : 4

TITLE OF THE PAPER : OBJECT ORIENTED PROGRAMMING WITH C++

Pedagogy	Hours	Lecture	Peer Teaching	GD/VIDOES/TUTORIAL	ICT
	4	3	0 / 1	0 / 1	0 / 1
PREAMBLE: This course introduces the object oriented concept with C++ Language.					
COURSE OUTCOME At the end of the Semester, the Students will be able to					Hrs P/S
UNIT 1	CO1: Define the basics of Object Oriented Programming				8
UNIT 2	CO2: Explain and apply concepts of objects, arrays, functions and constructors within a class				15
UNIT 3	CO3: Demonstrate ability to implement overloading and inheritance				18
UNIT 4	CO4: Apply and utilize the concept of Pointers, Virtual Functions and I/O Operations				10
UNIT 5	CO5: List the various File stream operations and Templates				9

Programme : B. Sc Computer Science

Semester II

Sub. Code : U22CS4

Part III : Core

Hours : 4 P/W 60 Hrs P/S

Credits : 4

TITLE OF THE PAPER: OBJECT ORIENTED PROGRAMMING WITH C++

UNIT I : Basic Concepts of Object Oriented Programming - Benefits OOP - Object oriented Languages - Applications of OOP - Structure Of C++ program -Tokens - Keywords - Identifiers and constants -Basic data types - User defined data types - Derived data types - Symbolic constants - Type compatibility - Declaration of variables - Dynamic Initialization of variables - reference variables - operators in C++ - Manipulators - Type cast operators - Expressions and their types -Implicit - Control Structures - The main function - function prototyping - Inline function - Function overloading

UNIT II : Specifying a class – Defining member function – Making an outside function inline – Nesting of member functions – Private member functions – Array within a class – Memory allocation for objects -Static data members- Static member functions – Array of objects – Objects as function arguments – Friendly functions – Returning objects – Constant member functions – Constructors – Parameterized constructor – Multiple constructors in a class – Constructors with default arguments – Dynamic initialization of objects – Copy constructor – Destructors.

UNIT III : Defining operator overloading – Overloading unary operators – Overloading binary operators – Overloading binary operators using friend function – Rules for overloading operators – Defining derived classes – Single inheritance – Making a private member inheritable – Multilevel inheritance – Multiple inheritance – Hierarchical inheritance – Hybrid inheritance – Virtual base classes
– Constructors in derived class – Member classes: Nesting of classes.

UNIT IV : Pointer to objects – this pointer – Pointers to derived classes – Virtual functions – Pure virtual functions – C++ Stream classes – Unformatted I/O operations – Managing output with manipulators.

UNIT V : Classes of file stream operations – Opening and closing files – Detecting end of file – More about open() function – File modes, File pointers and their manipulation – Sequential input and output operations – Command line arguments – Templates: Class templates and function templates.

TEXT BOOK :

1. Object Oriented Programming with C++, E. Balagurusamy ,MCGraw Hill Education (India) Private Limited, New Delhi, Sixth Edition – 2013.

CHAPTERS:

- UNIT I : CHAPTER 1 (EXCEPT 1.3, 1.4), CHAPTER 2 (ONLY 2.6), CHAPTER 3 (EXCEPT 3.20, 3.21, 3.22) AND CHAPTER 4
- UNIT II : CHAPTER 5 (EXCEPT 5.18,5.19), CHAPTER 6 (EXCEPT 6.8, 6.9, 6.10)
- UNIT III : CHAPTER 7 & CHAPTER 8
- UNIT IV : CHAPTER 9 & CHAPTER 10
- UNIT V : CHAPTER 11 (EXCEPT 22.8) AND CHAPTER 12(ONLY 12.2, 12.3, 12.4)

REFERENCE BOOKS:

1. C++ - The Complete Reference, Herbert Schildt, TMH, 1998.
 2. C++ How to Program, Paul Deital, Harvey DEital, PHI, Ninth Edition (2014).
 3. Ashok N. Kamthane, Object Oriented Programming with ANSI & turbo C++, Pearson Education,2006.
- Object Oriented Programming Using C++, Ashok Kumar Jagadev, Amiya Kumar Rath and Satchidananda Dehuri, PHI, New Delhi, 2007.

Programme : B. Sc Computer Science

Semester II

Sub. Code : U22CS4

Part III : Core

Hours : 4 P/W 60 Hrs P/S

Credits : 4

UNITS	TOPIC	LECTURE HOURS	MODE OF TEACHING
UNIT I			
1.	What is object oriented programming? Application of OOP	1	Lecture
2.	The Structure of C++ program - Different data types -Declaration and Initialization of variables – how to reference variables ?	2	Lecture
3.	Concepts of Operators, Expressions and Control Structures	2	Lecture
4.	Why we need main function ? Different function prototypes available and insight on concept of Function overloading	2	Lecture
5.	Discussion	1	Group Discussion
UNIT II			
6.	Specification of a class and Defining member function	1	Lecture
7.	Different access specifiers for member variables and Functions	1	Lecture
8.	Define an array within a class and Memory allocation for objects	1	Lecture
9.	Discuss about Static data members and member functions	1	Lecture
10.	Construction of program using previous concepts	1	Group Discussion
11.	Examples for function types	1	Peer Teaching
12.	Define Array of objects – Passing arguments to functions Purpose of Friend function – How to return objects to a class and concept of Constant member functions	3	Lecture
13.	Exercise Programs	1	Tutorial
14.	Introduction of Constructors and different types of constructor	2	Lecture
15.	Dynamic initialization of objects, Copy constructor and purpose of Destructors.	2	Lecture
16.	Overview of Unit II	1	ICT (NPTEL Videos)

UNIT III			
17.	Defining operator overloading with Syntax	1	Lecture
18.	Explaining different types of operator overloading - unary operators overloading and binary operators overloading	3	Lecture
19.	Implementation of binary operators using friend Function	2	Lecture
20.	Rules for overloading operators	1	Lecture
21.	Exercise Problems	1	Tutorial
22.	Defining derived classes and Inheritance with suitable Examples	1	Lecture
23.	Implementations of Single Inheritance and Making a private member inheritable	1	Lecture
24.	Implementation of Multilevel Inheritance with suitable examples	1	Lecture
25.	Implementation of Multiple Inheritance with suitable examples	1	Lecture
26.	Implementation of Hierarchical Inheritance with suitable examples	1	Lecture
27.	Implementation of Hybrid Inheritance with suitable examples	1	Lecture
28.	Exercise Programs	1	ICT (NPTEL Notes)
29.	Utilization of Constructors in derived class	1	Lecture
30.	Concept of nesting classes.	1	Lecture
31.	Applications	1	Group Discussion
UNIT IV			
32.	Introduction of Pointers	1	Lecture
33.	Concepts of Pointer to Object and Purpose of this pointer	1	Lecture
34.	Implementation of Pointers to a derived class	1	Lecture
35.	Exercise Problems	1	Tutorial
36.	Need of Virtual Functions – Defining the virtual functions – Concepts of Pure virtual Functions	2	Lecture
37.	Introduction of stream classes and implementation of unformatted I/O operations	1	Lecture
38.	Managing Output with manipulators	1	Lecture
39.	Efficient use of Pointers inside the class	1	ICT (NPTEL Notes)
40.	Applications	1	Group Discussion

UNIT V			
41.	Introduction of File Stream Operations – Opening, Closing and detecting end of files	2	Lecture
42.	More about open() function – File modes, File pointers and their manipulation	2	Lecture
43.	Manipulations in files using different operations	1	Peer Teaching
44.	Sequential input and output operations – Command line arguments	1	Lecture
45.	Introduction of Templates	1	Lecture
46.	Class templates and function templates	1	Lecture
47.	Overview about files	1	ICT (NPTEL Notes)

Course Outcomes (COs)	Programme Outcomes (POs)					Programme Specific Outcomes (PSOs)					Mean Scores of Cos
	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3	PSO4	PSO5	
CO1	2	3	3	3	3	3	2	4	3	4	3.00
CO2	3	3	4	4	3	4	3	3	3	3	3.30
CO3	4	3	3	3	4	4	3	4	3	3	3.40
CO4	4	3	3	3	3	4	3	3	4	3	3.30
CO5	3	3	3	3	4	3	3	3	4	3	3.20
Mean Overall Score											3.24

Result: The Score for this Course is 3.24 (High Relationship)

COURSE DESIGNER : Dr. M. SUMATHI
Associate Professor / PG and Research Department of Computer Science.

Programme : B. Sc. Computer Science

Semester II

Sub. Code : U22CS6

Part III :Core

Hours : 4 P/W 60Hrs P/S

Credits : 4

TITLE OF THE PAPER: DATA STRUCTURES AND ALGORITHMS

Pedagogy	Hours	Lecture	Peer Teaching	GD/VIDEOS/TUTORIAL	ICT
	4	4/3	0 / 1	0 / 1	0 / 1
PREAMBLE: To impart the fundamental knowledge about data and its organization, and problem solving approaches in the computing environment using data structures.					
COURSE OUTCOME					Hrs P/S
At the end of the Semester, the Students will be able to					
UNIT 1	CO1: Define algorithm and describes a specific algorithmic approach and apply it for solving common sorting and searching techniques.				12
UNIT 2	CO2: Explain the simple and common data structures – array , list, stack and queue. Solve the problems using these data structures..				12
UNIT 3	CO3: List the different types of linked list and its application				12
UNIT 4	CO4: Outline the advanced data structure – binary tree, its representation and discuss the application.				12
UNIT 5	CO5: Make Use of graph data structure and Classify its application.				12

Programme : B. Sc. Computer Science

Semester II

Sub. Code : U22CS6

Part III :Core

Hours : 4 P/W 60Hrs P/S

Credits : 4

TITLE OF THE PAPER: DATA STRUCTURES AND ALGORITHM

UNIT I : Introduction: Algorithm – Writing structured program, Analyzing algorithm- Divide and conquer – The general method, Binary search, Merge sort, Quick sort, Selection sort.

UNIT II : Introduction: Arrays – Ordered List – Sparse Matrices -Representation of Arrays- Stacks and Queues – A Mazing problem – Evaluations of expressions.

UNIT III : Linked list: Singly linked list – linked stacks and queues – polynomial addition – more on linked list – equivalence relations – doubly linked list.

UNIT IV: Trees: Basic terminology – binary trees – binary tree representations – binary tree traversal – threaded binary trees – binary tree representation of trees – application of trees -SetRepresentation.

UNIT V : Graphs: Terminology and representations – traversals, spanning trees – shortest paths.

TEXT BOOK(S):

1. Fundamentals of Data structures – Ellis Horowitz, Sartaj Sahani, Galgotia Pub., 1998.(Exclude analysis & Lemma)
2. Fundamentals of Computer Algorithms – Ellis Horowitz, Sartaj Sahni, Sangutheva Rajasekaran, Universities Press – 2nd Edition, 2008.(Exclude analysis & Lemma)

CHAPTERS:

UNIT I : CHAPTERS: 1
UNIT II : CHAPTERS: 2, 3.1 – 3.3
UNIT III : CHAPTERS: 4.1, 4.2, 4.4 – 4.8
UNIT IV : CHAPTERS: 5.1 – 5.7, 5.8.1
UNIT V : CHAPTERS: 6.1 – 6.3

CHAPTERS:

UNIT I : Chapter: 3.1, 3.3, 3.5 – 3.7

REFERENCE BOOKS

1. Pascal Plus Data Structures Algorithm And AdvanceD Programming – Nell.Dale, Susan.C.Lilly, TMH pub., 1985.
2. C and Data Structures – by P.Radha Ganesan, Scitech Pub, India Pvt Ltd.

Programme : B. Sc. Computer Science

Part III :Core

Semester II

Hours : 4 P/W 60Hrs P/S

Sub. Code : U22CS6

Credits : 4

UNITS	TOPIC	LECTURE HOURS	MODE OF TEACHING
UNIT 1			
1.	Introduction on the importance of Data Structures. Define algorithm. Explain how to write algorithm with a simple example.	1	Lecture
2.	Assignment on how to write algorithm for simpler problems.	1	Tutorial
3.	Programming Languages- Structured programming explanation. An overview of analysis	1	Lecture
4.	Discussion on various program solving approaches. Explain divide and conquer method with an example	1	Lecture
5.	Explain sequential and binary search methods. Discuss the relative merits and demerits	1	Lecture
6.	Describe recursion. Explain the function with a walk-through on the execution of a recursive program	1	Lecture
7.	Describe the logic and implementation of Quick Sort method	1	Lecture
8.	Describe the logic and implementation of Merge Sort method	1	Lecture
9.	Describe the logic and implementation of Selection Sort method	1	Lecture
10.	A tutorial session on how to write a program from the algorithmic steps.	1	Tutorial
11.	Give exercise to implement sorting and searching methods in C language.	1	Peer Teaching
12.	Conduct test on the understanding of various methods and techniques learned. Summarizing the concepts discussed.	1	Quiz & Discussion
UNIT 11			
13.	Describe array and its internal storage representation. Walk-through on how does system locate the elements	1	Lecture
14.	Define list and ordered list. Give exercise problems to demonstrate the application of list.	1	Lecture
15.	Describe the sparse matrix.	1	Lecture
16.	Discuss the operations (create, copy, transpose) on sparse matrix.	1	Lecture
17.	Describe Stack data structure and its implementation	2	Lecture

18.	Describe Queue data structure and its implementation	2	Lecture
19.	Discuss the Maze problem and its solution.	1	Lecture
20.	Describe the evaluation of expression - infix, prefix and postfix notation.	1	Lecture
21.	Describe the algorithm to convert from infix to postfix notation.	1	Lecture
22.	Conduct oral test and summarize the concepts discussed in Unit II	1	Group Discussion
UNIT III			
23	Describe Linked list and its advantages.	1	Lecture
24.	Implementation of singly linked list	1	Lecture
25.	Circularly linked list	1	Lecture
26.	Linked Stack and Queue	1	Lecture
27.	Solving Polynomial addition problem using linked list	2	Lecture
28.	Describing equivalence relation and its application in memory usage optimization	1	Lecture
29.	Solving equivalence relation problem using linked list	1	Lecture
30.	Describe doubly linked list and its advantages over singly linked list	1	Lecture
31.	Exercise problems on inserting and removing elements in singly linked list and doubly linked list	1	Tutorial
32.	Memory management using linked list	1	Lecture
33.	Assignment on solving exercise problems using linked list	1	Peer Teaching
UNIT IV			
34.	Tree terminology – Definition – an overview of its application	1	Lecture
35.	Tree and Binary Tree. Advantages of tree over list	1	Lecture
36.	Tree traversal – in order, pre order and post order	1	Lecture
37.	Exercise on adding and removing elements in a binary tree	2	Tutorial
38.	Binary tree operations – copy, check for equality	2	Lecture
39.	Threaded binary tree	1	Lecture
40..	Exercise Problems	1	Peer teaching
41.	Set operation – union and find	1	Lecture
42.	Set representation using tree	1	Lecture
43.	Conduct test to evaluate understanding	1	Quiz

UNIT V			
44.	Graph Terminology – complete, connected, directed	1	Lecture
45.	Graph representation – adjacency matrix and adjacency list	1	Lecture
46.	Graph applications	1	ICT – online resources
47.	Graph traversal – Depth first and Breadth first	1	Lecture
48.	Exercise problems	1	Tutorial
49.	Minimum cost spanning tree and its implementation	2	Lecture
50.	Shortest path algorithm and its implementation	2	Lecture
51.	Finding Algorithmic complexity of the problems solved	2	Tutorial and peer teaching
52.	Summary on various concepts, methods and techniques applied	1	Group discussion

Course Outcomes (COs)	Programme Outcomes (POs)					Programme Specific Outcomes (PSOs)					Mean Scores of Cos
	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3	PSO4	PSO5	
CO1	3	4	3	3	2	3	4	4	3	3	3.20
CO2	4	4	4	3	2	4	4	4	3	3	3.50
CO3	4	4	4	3	2	4	3	4	3	3	3.4
CO4	4	4	4	3	2	4	4	4	3	3	3.6
CO5	4	4	4	3	2	3	3	3	4	3	3.4
Mean Overall Score											3.42

Result: The Score for this Course is 3.42 (High Relationship)

COURSE DESIGNER : Mrs. A S. BABY RANI
Associate Professor / PG and Research Department of Computer Science

Programme :B. Sc Computer Science

Part III :Allied

Semester II

Hours : 4 P/W 60Hrs P/S

Sub. Code :U22ASS2

Credits :4

TITLE OF THE PAPER: MATHEMATICAL FOUNDATIONS

Pedagogy	Hours	Lecture	Peer Teaching	GD/VIDOES/TUTORIAL	ICT
	4	3	0 / 1	0 / 1	0 / 1
PREAMBLE: To impart the knowledge on the fundamental Mathematical concepts – Sets, Relations, Graph and Trees, which are the foundation of Logic, various representations of Problem Domain and operations.					
COURSE OUTCOME					Hrs P/S
At the end of the Semester, the Students will be able to					
UNIT 1	CO1: Recall the sets, operations on sets, verifying basic law using Venn Diagrams				8
UNIT 2	CO2: Demonstrate the relationship between sets, operations on relations, Representing the relationships with Hasse Diagrams and finding closure using Warshalls Algorithm; To have introduction to Lattices				13
UNIT 3	CO3: Build the knowledge on driving Truth Tables, equivalence of formulas and Quantifiers.				14
UNIT 4	CO4: Outline the concept of graph and basic terminologies, theorems.				16
UNIT 5	CO5: Show the special types of Graph – Tree and basic terminologies, essential theorems on Trees.				9

Programme :B. Sc Computer Science

Semester II

Sub. Code : U22ASS2

Part III :Allied

Hours : 4 P/W 60Hrs P/S

Credits :4

TITLE OF THE PAPER: MATHEMATICAL FOUNDATIONS

UNIT I

Set Theory: Introduction – Sets – Notation and Description of Sets – Subsets – Venn – Euler Diagrams – Operation on sets – Properties of set operations – Verification of basic laws and algebra by Venn diagram.

UNIT II

Relations and Functions: Relations – Representation of a relation – Operations on relations – equivalence relation – Closures & Warshalls Algorithm – Partial order Relation – Hasse Diagrams – Lattices.

UNIT III

Logic: Introduction – IF statements – Connectives – Truth table of a formula – Tautology – Tautological implications and Equivalence of formulae – Quantifiers.

UNIT IV

Graph Theory: Introduction- path and circuits: Isomorphism – Subgraphs- Walks, Paths and circuits- Connected graphs, Disconnected Graphs and components.

UNIT V

Euler Graphs: Hamiltonian Paths and circuits- Trees: Some properties of Trees – Distances and centers.

TEXT BOOK(S):

1. Discrete Mathematics – M.Venkataraman, N.Sridharan and N.Chandrasekaran – TheNational Publishing Company, May 2009.
2. Graph Theory – Narsingh Deo, Prentice Hall of India, 2002(UNIT IV & V)

CHAPTERS:

UnitI	: Chapter 1.1 to 1.8
Unit II	: Chapter 2 (2.2 to 2.6), 10.1
Unit III	: Chapter 9 (9.1 to 9.3, 9.6 to 9.8, 9.15)
UNIT IV	: Chapters 1, 2 - 2.1, 2.2, 2.4, 2.5 (Text Book 2)
UNIT V	: Chapters 2 - 2.6, 2.9, 3 - 3.1, 3.2, 3.4 (Text Book 2)

REFERNCE BOOK:

1. Discrete Mathematics for Computer Science, J. Truss, 2nd Edition Addison Wesley, Boston, 1999.

Programme :B. Sc Computer Science

Semester II

Sub. Code : U22ASS2

Part III :Allied

Hours : 4 P/W 60Hrs P/S

Credits :4

TITLE OF THE PAPER: MATHEMATICAL FOUNDATIONS

UNITS	TOPIC	LECTURE HOURS	MODE OF TEACHING
UNIT 1			
1.	Introduction to Sets	1	Lecture
2.	Notation and Description of Sets , Subsets , Venn – Euler Diagrams and Operation on sets	2	Lecture
3.	Solving problems on sets and representation in Venn diagram	1	Peer Teaching
4.	Properties of set operations	1	Lecture
5.	Verification of basic laws and algebra by Venn diagram.	1	Lecture
6.	Additional Problems on Sets, basic law verification	1	Lecture
7.	Applications of Sets in Real world problems	1	Group Discussion
UNIT 11			
8.	Relations : Representation of a relation.	1	Lecture
9.	Operations on relations	1	Lecture
10.	Equivalence relation	1	Lecture
11.	Solving problems on equivalence relation	1	Tutorial
12.	Closures	1	Lecture
13.	Warshalls Algorithm	1	Lecture
14.	Problem Solving using Warshalls Algorithm	1	Peer Teaching
15.	Partial order Relation	1	Lecture
16.	Application of Partial Order Relation	1	ICT (UGCEMIRC Video)
17.	Hasse Diagrams	1	Lecture
18.	Problem solving using Hasse Diagram	1	Lecture
19.	Lattices	1	Lecture
20.	Overview of Unit II	1	Presentation
UNIT III			
21.	Introduction to Logic: IF statements	1	Lecture
22.	Connectives	1	Lecture
23.	Problems using connectives	1	Group Discussion
24.	Writing Truth table of a formula	1	Tutorial

25.	Solving Exercise Problems	1	Peer Teaching
26.	Tautology	1	Lecture
27.	Solving Problems based on tautology	2	Lecture
28.	Tautological implications and Equivalence of formulae	1	Tutorial
29.	Problem solving : implications and equivalence of formulas	1	Demonstration
30.	Quantifiers	2	Lecture
31.	Solving exercise problems	1	Lecture
32.	Summary of Unit-II	1	Lecture
UNIT IV			
33.	Introduction to Graph	1	Lecture
34.	Real Time Problems in application to Graph	1	Group Discussion
35.	Path and circuits	1	Lecture
36.	Isomorphism	1	Lecture
37.	Subgraphs	1	Lecture
38.	Solving problems based on path, isomorphism and subgraphs	1	Peer Teaching
39.	Walks, Paths and circuits, Walk Vs Path Vs Circuit	1	Lecture
40.	Theorems on Walk, Path and Circuit	1	Lecture
41.	Solving exercise problems on walk, path and circuit	1	Tutorial
42.	Connected graphs	1	Lecture
43.	Solving problems – testing of Connected graph	1	Lecture
44.	Disconnected Graph	1	Lecture
45.	Components	1	Lecture
46.	Solving problem on components	1	Tutorial
47.	Overview of Unit-IV	2	Lecture
UNIT V			
48.	Hamiltonian Paths and circuits	1	Lecture
49.	Hamiltonian Paths VS circuits	1	Lecture
50.	Solving exercise problems on Hamiltonian path, circuit	1	Peer Teaching
51.	Introduction to Trees	1	Lecture
52.	Some properties of Trees, related theorems	1	Lecture
53.	Distances and related theorems	1	Lecture
54.	Finding the center(s) of tree	1	Lecture
55.	Solving Exercise Problems on Trees	1	Tutorial
56.	Overview of UNIT-V	1	ICT-NPTEL Video

Course Outcomes (COs)	Programme Outcomes (POs)					Programme Specific Outcomes (PSOs)					Mean Scores of Cos
	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3	PSO4	PSO5	
CO1	3	4	3	3	2	3	4	4	3	3	3.20
CO2	4	4	4	3	2	4	4	4	3	3	3.50
CO3	4	4	4	3	2	4	3	4	3	3	3.4
CO4	4	4	4	3	2	4	4	4	3	3	3.6
CO5	4	4	4	3	2	3	3	3	4	3	3.4
Mean Overall Score											3.42

Result: The Score for this Course is 3.42 (High Relationship)

COURSE DESIGNER: Mrs. G. SUDHA
Assistant Professor / PG and Research Department of Computer Science.

Programme :B. Sc Computer Science

Part III : Core

Semester II

Hours : 4 P/W 60Hrs P/S

Sub. Code : U22CS5P

Credits : 3

TITLE OF THE PAPER : PROGRAMMING IN C++ LAB

1. Printing Prime numbers between two given numbers.
2. Printing 3 digit numbers as a series of words. (*Ex. 543 should be printed out as FiveFour Three*).
3. Finding area of geometric shapes using function overloading.
4. Inline functions for simple arithmetic operations.
5. Demonstrating the use of Pre-defined Manipulators.
6. Demonstrating the use of friend function.
7. Creating student mark list using array of objects,
8. Demonstrating constructor overloading.
9. Overloading the unary – operator.
10. Demonstrating single inheritance.
11. Demonstrating the use of this pointer.
12. Designing our own manipulator.
13. Illustrating function templates.
14. Illustrating class templates.
15. Overloading the binary + operator.
16. Demonstrating Multiple inheritance.
17. Demonstrating Multilevel inheritance.
18. Demonstrating Hierarchical inheritance.
19. Demonstrating Virtual functions.
20. Processing mark list using binary file.
21. Count number of objects in a file.
22. Demonstrating the use of Command-line arguments.