NARAYANA ENGINEERING COLLEGE::GUDUR

AUTONOMOUS

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

B.Tech – CSE - Course Structure, w.e.f AY: 2023-24

DEPARTMENT VISION & MISSION

VISION OF THE DEPARTMENT

- To produce globally competent software professionals in the field of computer science and engineering to meet the needs of industry and society along with research and consultancy, lifelong learning, leadership qualities and ethics.

MISSION OF THE DEPARTMENT

- To deliver quality technical education by practicing innovative teachinglearning processes making student's self-sufficient individuals
- To inculcate innovative thinking and problem solving skills in learnersthrough training programs and collaborative interaction with industry.
- To develop professional behaviour with strong ethical values, leadershipqualities and lifelong learning by providing value based education

PEOs, POs, PSOs

PEOs

PEO 1: To attain higher position in career by exhibiting expertise in solving real world problems.

PEO 2: Fill technical gaps and take leadership roles and achieve substantive resultsfor the development of organization.

PEO 3: Adapt to rapidly changing technologies through lifelong learning.

POs

1. Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

2. Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

3. Design/development of solutions: Design solutions for complex engineering 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 considerations.

4. Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

5. Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

6. The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. **Individual and team work**: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

10. **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.

11. **Project management and finance**: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work,as a member and leader in a team, to manage projects and in multidisciplinary environments.

12: **Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

PSOs

PSO 1: Software Product Development: Apply the principles and practices of software Engineering for developing quality software applications

PSO 2: Employment: Get employed in industries through their knowledge attained in Basic and advanced programming languages, specialized software packages or become an entrepreneur.

\

SEMESTER - I

Course Code	Category	Course Title	Contact Periods per week				Credits	Scheme of Examination Max. Marks		
			L	T	P	Total		Int. Marks	Ext. Marks	Total marks
21MA1001	BS	Algebra and Calculus	3	1	0	4	4	40	60	100
21CH1001	BS	Chemistry	3	0	0	3	3	40	60	100
21ES1001	ES	Problem Solving and Programming	3	0	0	3	3	40	60	100
21EN1001	HS	English	2	0	0	2	2	40	60	100
21CH1501	BS	Chemistry Lab	0	0	3	3	1.5	40	60	100
21ES1503	ES	Engineering Graphics	0	1	4	5	3	40	60	100
21ES1501	ES	Problem Solving and Programming lab	0	0	3	3	1.5	40	60	100
21EN1501	HS	English Language Lab	0	0	3	3	1.5	40	60	100
21CS8101	MC	Mandatory course I:Induction Program	--							
		Counselling/Mentoring	0	0	1	1	0	--	--	--
		Sports/Hobby Clubs/Activities	0	0	2	2	0	--	--	--
		Activity Point Programme	During the Semester				20 Pts			
			11	**2**	**16**	**29**	**19.5**	**320**	**480**	**800**

SEMESTER -II

Course Code	Category	Course Title	Contact Periods per week				Credits	Scheme of Examination Max. Marks		
			L	T	P	Total		Int. Marks	Ext. Marks	Total marks
21MA1002	BS	Probability and Statistics	3	1	0	4	4	40	60	100
21PH1004	BS	Semiconductor Physics	3	0	0	3	3	40	60	100
21ES1004	ES	Basic Electrical and Electronics Engineering	3	0	0	3	3	40	60	100
21ES1005	ES	Python Programming and Data Science	3	0	0	3	3	40	60	100
21PH1504	BS	Semiconductor physics lab	0	0	3	3	1.5	40	60	100
21ES1507	ES	Basic Electrical and Electronics Engineering lab	0	0	2	2	1	40	60	100
21ES1505	ES	Engineering and IT Workshop	0	0	3	3	1.5	40	60	100
21ES1508	ES	Python Programming and Data Science Lab	0	0	3	3	1.5	40	60	100
21EN1502	HS	Communication Skills Lab	0	0	2	2	1	40	60	100
21MC8102-13	MC	Mandatory Course II	2	0	0	2	0	--	--	--
		Counselling/Mentoring	0	0	1	1	0	--	--	--
		Sports/Hobby Clubs/Activities	0	0	2	2	0	--	--	--
		Activity Point Programme	During the Semester				20 Pts			
			14	**1**	**16**	**31**	**19.5**	**360**	**540**	**900**

SEMESTER – III

Course Code	Category	Course Title	Contact Periods per week				Credits	Scheme of Examination Max. Marks		
			L	T	P	Total		Int. Marks	Ext. Marks	Total marks
21EN1002	HS	Universal Human Values	3	0	0	3	3	40	60	100
21ES1009	ES	Data Structures and Algorithms	3	0	0	3	3	40	60	100
21CS2001	PC	Computer Organization and Architecture	3	0	0	3	3	40	60	100
21CS2002	PC	Database Management Systems	3	0	0	3	3	40	60	100
21CS2003	PC	Mathematical Foundation for Computer Science	3	0	0	3	3	40	60	100
21CS2004	PC	Object Oriented Programming through Java	3	0	0	3	3	40	60	100
21ES1513	ES	Data Structures and Algorithms Lab	0	0	3	3	1.5	40	60	100
21CS2501	PC	Database Management Systems Lab	0	0	3	3	1.5	40	60	100
21CS2502	PC	Object Oriented Programming through Java Lab	0	0	3	3	1.5	40	60	100
21CD6001	SC	Career competency development I	0	0	2	2	1	40	60	100
21CC6001	SC	Value added course/Certificate course I	0	0	0	0	1	40	60	100
		Counselling/Mentoring	0	0	1	1	0	--	--	--
		Sports/Hobby Clubs/Activities	0	0	2	2	0	--	--	--
		Activity Point Programme	During the Semester				20 Pts			
			18	**0**	**14**	**32**	**24.5**	**440**	**660**	**1100**

SEMESTER –IV

Course Code	Category	Course Title	Contact Periods per week				Credits	Scheme of Examination Max. Marks		
			L	T	P	Total		Int. Marks	Ext. Marks	Total marks
21MA1007	BS	Exploratory Data Analysis with R	3	0	0	3	3	40	60	100
21CS2005	PC	Computer Networks	3	0	0	3	3	40	60	100
21CS2006	PC	Operating Systems	3	0	0	3	3	40	60	100
21CS2007	PC	Software Engineering	3	0	0	3	3	40	60	100
21EC3011	OE	Open Elective I(Digital Logic Design)	3	0	0	3	3	40	60	100
21MA1501	BS	Exploratory Data Analysis with R Lab	0	0	3	3	1.5	40	60	100
21CS2503	PC	Operating Systems and Computer Networks Lab	0	0	3	3	1.5	40	60	100
21CS2504	PC	Software Engineering Lab	0	0	3	3	1.5	40	60	100
21CD6002	SC	Career Competency development II	0	0	2	2	1	40	60	100
21IC6001	SC	Industry oriented Course-I	0	0	0	0	1	100	--	100
21MC8007	MC	Mandatory course III(Leader ship & Management Course)	2	0	0	2	0	--	--	--
		Counselling/Mentoring	0	0	1	1	0	--	--	--
		Sports/Hobby Clubs/Activities	0	0	2	2	0	--	--	--
		Activity Point Programme	During the Semester				20 Pts			
			17	**0**	**14**	**31**	**21.5**	**460**	**540**	**1000**

SEMESTER –V

Course Code	Category	Course Title	Contact Periods per week				Credits	Scheme of Examination Max. Marks		
			L	T	P	Total		Int. Marks	Ext. Marks	Total marks
21CS2008	PC	Artificial Intelligence	3	0	0	3	3	40	60	100
21CS2009	PC	Design and Analysis of Algorithms	3	0	0	3	3	40	60	100
21CS2010	PC	Theory of Computation	3	0	0	3	3	40	60	100
21EC3007	OE	Open Elective II (Microprocessors and Micro Controllers)	3	0	0	3	3	40	60	100
21CS4002	PE	Professional Elective I(Software Project Management)	3	0	0	3	3	40	60	100
21CS2505	PC	Artificial intelligence lab	0	0	2	2	1	40	60	100
21CS2506	PC	Coding Lab-1	0	0	2	2	1	40	60	100
21CS2507	PC	Design and Analysis of Algorithms Lab	0	0	2	2	1	40	60	100
21CD6003	SC	Career competency development III	0	0	2	2	1	40	60	100
21CC6002	SC	Value added Course/Certificate Course II	0	0	0	0	1	40	60	100
20CS7501	PR	Internship I/On job Training/Comm. Service Project	0	0	0	0	1.5	40	60	100
		Counselling/Mentoring	0	0	1	1	0	--	--	--
		Sports/Hobby Clubs/Activities	0	0	2	2	0	--	--	--
		Activity Point Programme	During the Semester				20 Pts			
			15	**0**	**11**	**26**	**21.5**	**440**	**660**	**1100**

SEMESTER –VI

Course Code	Category	Course Title	Contact Periods per week				Credits	Scheme of Examination Max. Marks		
			L	T	P	Total		Int. Marks	Ext. Marks	Total marks
21HS5001	HS	Humanities and Social Science Elective (Managerial Economics & Financial Analysis)	2	0	0	2	2	40	60	100
21CS2011	PC	Machine Learning	3	0	0	3	3	40	60	100
21CS2012	PC	Web Technologies	3	0	0	3	3	40	60	100
21EC3006	OE	Open elective III (Internet of Things)	3	0	0	3	3	40	60	100
21CS4007	PE	Professional elective II (Software Architecture)	3	0	0	3	3	40	60	100
21CS4014	PE	Professional Elective III (Cloud Computing)	3	0	0	3	3	40	60	100
21CS2508	PC	Machine Learning Lab	0	0	2	2	1	40	60	100
21CS2509	PC	Web Technologies Lab	0	0	3	3	1.5	40	60	100
21CD6004	SC	Career competency Development IV	0	0	2	2	1	40	60	100
21IC6002	SC	Industry oriented Course-II	0	0	0	0	1	100	--	100
21MC8008	MC	Mandatory course IV (Human Relations at Work)	2	0	0	2	0	--	--	--
		Counselling/Mentoring	0	0	1	1	0	--	--	--
		Sports/Hobby Clubs/Activities	0	0	2	2	0	--	--	--
		Activity Point Programme	During the Semester				20 Pts			
			19	**0**	**10**	**29**	**21.5**	**460**	**540**	**1000**

SEMESTER –VII

Course Code	Category	Course Title	Contact Periods per week				Credits	Scheme of Examination Max. Marks		
			L	T	P	Total		Int. Marks	Ext. Marks	Total marks
21CS2013	PC	Cryptography and Network Security	3	0	0	3	3	40	60	100
21CS2014	PC	Mobile Application Development	3	0	0	3	3	40	60	100
21CS2015	PC	Deep Learning	2	0	0	2	2	40	60	100
	OE	Open Elective IV	3	0	0	3	3	40	60	100
21CS4016-20	PE	Professional Elective IV	3	0	0	3	3	40	60	100
21CS4021-25	PE	Professional Elective V	3	0	0	3	3	40	60	100
21CS2510	PC	Mobile Application Development Lab	0	0	2	2	1	40	60	100
21CS2511	PC	Deep Learning Lab	0	0	3	3	1.5	40	60	100
21CD6005	SC	Career Competency Development V	0	0	2	2	1	40	60	100
21CC6501	SC	Skill Development Training	0	0	2	2	1	40	60	100
20CS7502	PR	Internship II/On job Training/Comm. Service Project	0	0	0	0	1.5	40	60	100
		Counselling/Mentoring	0	0	1	1	0	--	--	--
		Sports/Hobby Clubs/Activities	0	0	2	2	0	--	--	--
		Activity Point Programme	During the Semester				20 Pts			
			17	**0**	**12**	**29**	**23**	**440**	**660**	**1100**

SEMESTER –VIII

Course Code	Category	Course Title	Contact Periods per week				Credits	Scheme of Examination Max. Marks		
			L	T	P	Total		Int. Marks	Ext. Marks	Total marks
20CS7503	PR	Project work, Seminar and internship	0	0	0	0	12	60	140	200
			0	**0**	**0**	**0**	**12**	**60**	**140**	**200**

OPEN ELECTIVES (OE)

OPEN ELECTIVES OFFERED BY DEPARTMENT OF CSE	
Course code	**TITLE OF THE COURSE**
21CS3001	Data Structures and Algorithms
21CS3002	Python Programming and Data Science
21CS3003	Object Oriented Programming through JAVA
21CS3004	Advanced Java Programming
21CS3005	Database Management Systems
21CS3006	Operating Systems
21CS3007	Computer Networks
21CS3008	Mobile Application Development
21CS3009	Web Technologies
21CS3010	Artificial intelligence
21CS3011	Cryptography and Network Security
21CS3012	Cloud Computing

THE PROFESSIONAL ELECTIVES

The Professional Elective Courses (PE) are shown in different tracks/groups: The students will have options of selecting the electives from the different tracks/groups depending on the specialization one wishes to acquire.

Electives Track/ Groups	**Professional Elective-1**	**Professional Elective-2**	**Professional Elective-3**	**Professional Elective-4**	**Professional Elective-5**
Computer Networks and Securities	Wireless Sensor Networks 21CS4001	Ethical Hacking 21CS4006	Information and Cyber Security 21CS4011	Computer Forensics 21CS4016	Block chain Technologies 21CS4021
Software Engineering	Software Project Management 21CS4002	Software Architecture 21CS4007	Software Testing 21CS4012	Object Oriented Analysis and Design 21CS4017	DEVOPS 21CS4022
Data Science and Engineering	Data warehousing and data mining 21CS4003	Business Intelligence and Analytics 21CS4008	**Data Virtualization Techniques** 21CS4013	**Reinforcement Learning** 21CS4018	Tools and Techniques for Data Science 21CS4023
Cloud Computing	Distributed Systems 21CS4004	Service Oriented Architecture 21CS4009	Cloud Computing 21CS4014	High Performance Computing 21CS4019	Cloud Security 21CS4024
Virtualization and Others	Game Development 21CS4005	**Big Data Analytics** 21CS4010	**Information Storage and Retrieval Systems** 21CS4015	Augmented and Virtual Reality 21CS4020	Virtualization Technologies 21CS4025
MOOCS	**MOOCS-1 21CS4026**	**MOOCS-2 21CS4027**	**MOOCS-3 21CS4028**	**MOOCS-4 21CS4029**	**MOOCS-5 21CS4030**

LIST OF HONOR SUBJECTS

Course code	Course Name	L-T-P	Credits
21CSH001	Secure Software Engineering	3-1-0	4
21CSH002	Multi-core Architecture & Programming	3-1-0	4
21CSH003	Reinforcement Learning	3-1-0	4
21CSH004	Trusted Network Systems	3-1-0	4
21CSH005	Parallel Database Systems	3-1-0	4

LIST OF MINOR SUBJECTS

Course code	Course Name	L-T-P	Credits
21CSM001	Design and Analysis of Algorithms	3-1-0	4
21CSM002	Database Management Systems	3-1-0	4
21CSM003	Computer Networks	3-1-0	4
21CSM004	Operating Systems	3-1-0	4
21CSM005	Artificial Intelligence	3-1-0	4

Humanities and Social Science Elective

S. NO	SUBJECT	CREDITS
1	Managerial Economics & Financial Analysis	3
2	Management Science	3
3	E-Business	3
4	Organizational Behaviour	3
5	Enterprise Resource Planning	3

HUMANITIES AND SOCIAL SCIENCES (HS)

SEMESTER	Course code	SUBJECT	CREDITS
I Sem	21EN1001	English	2
	21EN1501	English Language Lab	1.5
II Sem	21EN1502	Communication Skills Lab	1
III Sem	21EN1002	Universal Human Values	3
VI Sem	21HS5001-05	Humanities and Social Science Elective	2
		TOTAL	**9.5**

BASIC SCIENCES (BS)

SEMESTER	Course code	SUBJECT	CREDITS
I Sem	21MA1001	Algebra and Calculus	4
	21CH1001	Chemistry	3
	21CH1501	Chemistry Lab	1.5
II Sem	21MA1002	Probability and Statistics	4
	21PH1004	Semiconductor Physics	3
	21PH1504	Semiconductor physics lab	1.5
IV Sem	21MA1007	Exploratory Data Analysis with R	3
	21MA1501	Exploratory Data Analysis with R Lab	1.5
		TOTAL	**21.5**

ENGINEERING SCIENCES (ES)

SEMESTER	Course code	SUBJECT	CREDITS
I Sem	21ES1001	Problem Solving and Programming	3
	21ES1503	Engineering Graphics Lab	3
	21ES1501	Problem Solving and Programming lab	1.5
II Sem	21ES1004	Basic Electrical and Electronics Engineering	3
	21ES1005	Python Programming and Data Science	3
	21ES1507	Basic Electrical and Electronics Engineering lab	1
	21ES1505	Engineering and IT Workshop	1.5
	21ES1508	Python Programming and Data Science Lab	1.5
III Sem	21ES1009	Data Structures and Algorithms	3
	21ES1513	Data Structures and Algorithms lab	1.5
		TOTAL	**22**

PROFESSIONAL CORE (PC)

SEMESTER	Course code	SUBJECT	CREDITS
III Sem	21CS2001	Computer Organization and Architecture	3
	21CS2002	Database Management Systems	3
	21CS2003	Mathematical Foundation for Computer Science	3
	21CS2004	Object Oriented Programming through Java	3
	21CS2501	Database Management Systems lab	1.5
	21CS2502	Object Oriented Programming through Java Lab	1.5
IV Sem	21CS2005	Computer Networks	3
	21CS2006	Operating Systems	3
	21CS2007	Software Engineering	3
	21CS2503	Operating Systems and Computer Networks Lab	1.5
	21CS2504	Software Engineering Lab	1.5
V Sem	21CS2008	Artificial Intelligence	3
	21CS2009	Design and Analysis of Algorithms	3
	21CS2010	Theory of Computation	3
	21CS2505	Artificial intelligence lab	1
	21CS2506	Coding Lab	1
	21CS2507	Design and Analysis of Algorithms Lab	1
VI Sem	21CS2011	Machine Learning	3
	21CS2012	Web Technologies	3
	21CS2508	Machine Learning Lab	1
	21CS2509	Web Technologies Lab	1.5
VII Sem	21CS2013	Cryptography and Network Security	3
	21CS2014	Mobile Application Development	3
	21CS2015	Deep Learning	2
	21CS2510	Mobile Application Development Lab	1
	21CS2511	Deep Learning Lab	1.5
		TOTAL	**58**

PROFESSIONAL ELECTIVES (PE)

SEMESTER	Course code	SUBJECT	CREDITS
V Sem	21CS4001-05	Professional elective 1	3
VI Sem	21CS4006-10	Professional elective 2	3
	21CS4011-15	Professional elective 3	3
VII Sem	21CS4016-20	Professional elective 4	3
	21CS4021-25	Professional elective 5	3
		TOTAL	**15**

OPEN ELECTIVES (OE)

SEMESTER	Course code	SUBJECT	CREDITS
IV Sem	21EC3011	Open Elective 1(Digital Logic Design)	3
V Sem	21EC3007	Open Elective 2 (Microprocessors and Micro Controllers)	3
VI Sem	21EC3006	Open Elective 3 (Internet of Things)	3
VII Sem		Open Elective 4	3
		TOTAL	**12**

SKILL ORIENTED COURSES (SC)

SEMESTER	Course code	SUBJECT	CREDITS
III SEM	21CD6001	Career competency development I	1
	21CC6001	Value added course/Certificate course I	1
IV SEM	21CD6002	Industry oriented Course-I	1
	21IC6001	Career Competency development II	1
V SEM	21CD6003	Career competency development III	1
	21CC6002	Value added Course/Certificate Course II	1
VI SEM	21CD6004	Career competency Development IV	1
	21IC6002	Industry oriented Course-II	1
VII SEM	21CD6005	Career competency Development V	1
	21CC6501	Skill development Training	1
		TOTAL	10

PROJECT (PR)

SEMESTER		SUBJECT	CREDITS
V Sem	20CS7501	Internship I/On job Training/Comm. Service Project	1.5
VII Sem	20CS7502	Internship II/On job Training/Comm. Service Project	1.5
VIII Sem	20CS7503	Project work, Seminar and internship	12
		TOTAL	15

Credits Table

SUBJECT AREA	CREDITS PER SEMESTER								CREDITS
	I	II	III	IV	V	VI	VII	VIII	
HS	3.5	1	3	0	0	2	0	0	**9.5**
BS	8.5	8.5	0	4.5	0	0	0	0	**21.5**
ES	7.5	10	4.5	0	0	0	0	0	**22**
PC	0	0	15	12	12	8.5	10.5	0	**58**
OE	0	0	0	3	3	3	3	0	**12**
PE	0	0	0	0	3	6	6	0	**15**
PR	0	0	0	0	1.5	0	1.5	12	**15**
SC	0	0	2	2	2	2	2	0	**10**
TOTAL	**19.5**	**19.5**	**24.5**	**21.5**	**21.5**	**21.5**	**23**	**12**	**163**

SEMESTER-I

NARAYANA ENGINEERING COLLEGE::GUDUR								
21ES1001	PROBLEM SOLVING AND PROGRAMMING							R21
Semester	**Hours / Week**			**Total hrs**	**Credit**	**Max Marks**		
	L	**T**	**P**		**C**	**CIE**	**SEE**	**TOTAL**
I	3	0	0	48	3	40	60	100

Pre-requisite: Mathematics Knowledge, Analytical and Logical skills

Course Objectives:

- To understand various steps in Program development.
- To understand the basic concepts in C Programming Language.
- To learn how to write modular and readable C Programs.
- To learn the syntax and semantics of a C Programming language.
- To learn structured programming approach for problem solving.

Course Outcomes: After successful completion of the course, the student will be able to:

CO 1	Identify methods to solve a problem through computer programming. **(BL - 3)**
CO 2	Understand the use of operators and input/output. **(BL - 2)**
CO 3	Understand the difference and the usage of various control statements and Functions**(BL - 2)**
CO 4	Apply the Arrays and Pointers for solving problems. **(BL - 3)**
CO 5	Explain User-Defined Data Types and Files. **(BL - 2)**

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2			2							1	3	1
CO2	3	1			1								3	
CO3	3	1		1	2								3	1
CO4	3				1								1	
CO5	3		2		2							3	3	2
C06	3		2		2								3	1

COURSE CONTENT		
MODULE – 1	**Fundamentals of Computers and Programming**	**10 HOURS**
Introduction to Programming, Algorithms and Flowcharts: Programs and Programming, Programming languages, Compiler, Interpreter, Algorithms, Flowcharts, How to Develop a Program. **Basics of C:** Introduction**,** Character Set, Structure of a C Program, A Simple C Program, Variables, Data Types and Sizes, Declaration, Identifiers, Keywords, Constants, Assignment, and Initialization.		
At the end of the Module 1, students will be able to: **1.** Solve problems using language independent notations. **(BL - 3)** **2.** Understand the compilers and interpreters. **(BL - 2)** **3.** Understand Basic Structured of Programming in C. **(BL - 2)** **4.** Develop algorithms and flowcharts for problems.**(BL - 3)** **5.** Understand various Tokens in C language.**(BL - 2)**		
MODULE -2	**Operators and Input and Output**	**9 HOURS**
Operators and Expressions: Arithmetic Operators, Relational Operators, Logical Operators, Bitwise Operators, Conditional Operator, Comma operator, size of operator, Expressions, L values and R values, Expression Evaluation- Precedence and Associativity, Type Conversion. **Input and Output:** Basic Screen and Keyboard I/O in C, Formatted Input and Output, Unformatted Input and Output Functions**.**		
At the end of the Module 2, students will be able to: **1.** Illustrate the working of expressions.**(BL - 2)** **2.** Understand the precedence and Associativity rules of operators. **(BL - 2)** **3.** Understand the rules of type conversion. **(BL - 2)** **4.** Explain the Formatted and Unformatted I/O functions. **(BL - 2)**		
MODULE-3	**Control Statements and Functions**	**10 HOURS**
Control Statements: Selection Statements - if, Nested if, if-else, Nested if-else, else-if ladder, switch Looping Statements - while, do-while, for, Nested loops, Unconditional Statements - goto, break, Continue, return. **Functions:** Introduction, Using Functions, Passing Arguments to a Function, Working with Function, Scope and Extent, Recursion, The C Preprocessor, Storage classes		
At the end of the Module 3, students will be able to: **1.** Understand Selection Statements. **(BL - 2)** **2.** Understand Looping and Unconditional Statements. **(BL - 2)** **3.** Understand the basic concept of functions. **(BL - 2)** **4.** Understand concepts of Recursion, Preprocessor and storage classes. **(BL - 2)**		
MODULE-4	**Arrays and Pointers**	**10 HOURS**
Arrays and Strings: Introduction, One-Dimensional Array, Multidimensional Arrays, Passing Arrays to Function, Strings - Declaration, Initialization, Printing Strings, String Input, Character Manipulation, String Manipulation, Arrays of Strings. **Pointers:** Fundamentals, Pointer Declarations, Operations on pointers, Passing Pointers to a Function, Pointers and Arrays, Arrays of Pointers, Pointer to Pointer, Pointer to Functions, Command line arguments, Dynamic Memory Management.		
At the end of the Module 4, students will be able to: **1.** Understand the concept of Arrays. **(BL - 2)** **2.** Understand the concept of pointers. **(BL - 2)** **3.** Explain Dynamic Memory Management. **(BL -2)**		
MODULE-5	**User-Defined Data Types and Files**	**9 HOURS**
Structures and Unions: Basics of Structures, Nesting of Structures, Arrays of Structures, Structures and Pointers, Structures and Functions, Self-Referential Structures, Unions, Bit-fields, Enumerations, typedef. **Files:** Introduction, Using Data Files in C, Working with Text Files, Random Accesses to Files.		
At the end of the Module 5, students will be able to: **1.** Explain user defined data types like structures and unions. **(BL - 2)** **2.** Understand the concept of Self-Referential Structures. **(BL - 2)** **3.** Understand the working of files. **(BL - 2)**		
	Total hours:	**48 HOURS**
Content Beyond Syllabus: 1. Analysis of Algorithms 2. Text Vs. Binary Files 3. Variable Length Argument Lists		

Text Book(s):

1. Pradip Dey, and Manas Ghosh, "Programming in C", 2018, Oxford University Press.
2. Byron Gottfried, Schaum's Outline of Programming with C, 4th Edition, 2018, McGraw-Hill

Reference Books :

1. R.G. Dromey, "How to Solve it by Computer". 2014, Pearson.
2. Computer Fundamentals by Anita Goel, 2010, Pearson Publication
3. Brian W. Kernighan, and Dennis M. Ritchie, "The C Programming Language", 2ndEdition, Pearson.
4. Programming in C, 3/e : A Practical Approach by Ajay Mittal, Pearson Publication
5. C: The Complete Reference by SCHILDT and HERBERT, McGraw Hill, 4th Edition, 2020
6. Problem Solving with C by SOMASHEKARA, M. T., GURU, D. S., MANJUNATHA, K. S., PHI Learning, 2nd Edition, 2018
7. C How to Program,Paul Deitel, Deitel & Harvey Deitel, 6th Edition, Pearson Education
8. Programming in C and Data Structures, Jeri R. Hanly, Elliot B. Koffman, Ashok Kamthane and A.Ananda Rao, Pearson Education, 1st Edition, 2010.
9. C for Engineers and Scientists, H.Cheng, Mc.Graw-Hill International Edition Education / PHI, 2009
10. Programming in C – Stephen G. Kochan, 4th Edition, PearsonEductaion, 2015
11. Programming in ANSI in C, E Balaguruswamy, Tata McGraw Hill, 8th Edition, 2019
12. Computer Concepts and Programming in C, R.S. Salaria, Khanna Publishing, 2017
13. Let us C, Yashavant P. Kanetkar, BBP Publications, Delhi, 16th Edition, 2017

NARAYANA ENGINEERING COLLEGE::GUDUR

PROBLEM SOLVING AND PROGRAMMING LABORATORY

I year I Semester: Common to All

Course Code	Category	Hours / Week			Credits	Maximum Marks		
		L	T	P	C	CIA	SEE	Total
21ES1501		-	-	3	1.5	40	60	100
Contact Classes: Nil	**Tutorial Classes: Nil**	**Practical Classes: 36**				**Total Classes: 36**		

OBJECTIVES:

The course should enable the students to:

I. Formulate problems and implement algorithms using C programming language.
II. Develop programs using decision structures, loops and functions.
III. Learn memory allocation techniques using pointers.
IV. Use structured programming approach for solving of computing problems in realworld.

LIST OF EXPERIMENTS

Week-1 **OPERATORS AND EVALUATION OF EXPRESSIONS**

a. Write a C program to check whether a number is even or odd using ternaryoperator.
b. Write a C program to perform the addition of two numbers.
c. Write a C program to evaluate the arithmetic expression ((a + b / c * d - e) * (f - g)). Read the valuesa, b, c, d, e, f, g from the standard inputdevice.
d. Write a C program to find the sum of individual digits of a 3 digitnumber.
e. Write a C program to read the values of x and y and print the results of the following expressions in one line:
   i. (x + y) / (x - y)
   ii. (x + y)(x - y)

Week-2 **CONTROL STRUCTURES**

a. Write a C program to find the given year is leap or not
b. A Fibonacci sequence is defined as follows: The first and second terms in the sequence are 0 and 1. Subsequent terms are found by adding the preceding two terms in the sequence. Write a C program to generate the first n terms of the sequence.
c. Write a C program to generate all the prime numbers between 1 and n, where n is a value supplied by the user.
d. Write a C program to check largest number among three numbers

Week-3 **CONTROL STRUCTURES**

a. Write a C program, which takes two integer operands and one operator from the user, performs theoperation and then prints the result. (Consider the operators +, -, *, /, % and use switch statement).
b. Write a C program to calculate sum of n natural numbers
c. Write a C program to find the roots of a quadraticequation.
d. Write a C program to check whether a given 3 digit number is Armstrong number or not.
e. Write a C program to factorial value for a given positive integer

Write a C program to reverse the number

Week-4 **ARRAYS**

a. Write a C program to find the sum of n array elements.
b. Write a C program to perform thefollowing:
   i. Addition of two matrices
   ii. Multiplication of two matrices
c. Write a C program to count and display positive, negative, odd and even numbers in an array.

Week-5	STRINGS

a. Write a C program that uses functions to perform the following operations:
   i) String reverse
   ii) String length
   iii) String conversion
   iv) String copy
b. Write a C program to determine if the given string is a palindrome or not.
c. Write a C program that reads a line of text and counts all occurrence of a particular word.

Week-6	FUNCTIONS

a. Write C programs that use both recursive and non-recursive functions
   i. To find the factorial of a given integer.
   ii. To find the greatest common divisor of two given integers.
b. Write C programs that use both recursive and non-recursive functions
   i. To print Fibonacci series.
   ii. To solve towers of Hanoi problem.
c. Write a C program to print the transpose of a given matrix usingfunction.
d. Write a C program to call by value

Week-7	POINTERS

a. Write a C program to concatenate two strings using pointers.
b. Write a C program to find the length of string using pointers.
c. Write a C program to compare two strings usingpointers.
d. Write a C program to copy a string from source to destination using pointers.
e. Write a C program to pass pointers to a function(call-by-reference) .

Week-8	STRUCTURES AND UNIONS

a. Write a C program to compute the monthly pay of 100 employees using each employees name, basic pay. The DA is computed as 52% of the basic pay. Gross-salary (basic pay + DA). Print the employees name and gross salary.
b. Create a Book structure containing book_ id, title, author name and price. Write a C program to pass a structure as a function argument and print the bookdetails.
c. Create a union containing 6 strings: name, home_ address, hostel_ address, city, state and zip. Write a C program to display your present address.
d. Write a C program to define a structure named DOB, which contains name, day, month and year. Using the concept of nested structures display your name and date of birth.

Week-9	FILES

a. Write a C program to display the contents of afile.
b. Write a C program to copy the contents of one file to another.
c. Write a C program for fseek() function
d. Two files DATA1 and DATA2 contain sorted lists of integers. Write a C program to merge the contents of two files into a third file DATA
e. Write a C program to count the no. of characters present in the file.

REFERENCE BOOKS:

1. Yashavant Kanetkar, "Let Us C", BPB Publications, New Delhi, 13th Edition, 2012.
2. Oualline Steve, "Practical C Programming", OReilly Media, 3rd Edition, 1997.
3. King K N, "C Programming: A Modern Approach", Atlantic Publishers, 2nd Edition, 2015.
4. Kochan Stephen G, "Programming in C – A Complete Introduction to the C Programming Language",Sams Publishers, 3rd Edition, 2004.

Linden Peter V, "Expert C Programming: Deep C Secrets", Pearson India, 1st Edition, 1994.

SEMESTER-II

NARAYANA ENGINEERING COLLEGE::GUDUR							
21ES1005	PYTHON PROGRAMMING AND DATA SCIENCE						R21
Semester	Hours / Week			Total hrs	Credit	Max Marks	
	L	T	P		C	CIE	SEE	TOTAL
II	3	0	0	48	3	40	60	100

Pre-requisite: Basics of programming Language.

Course Objectives:

1. To learn about Python programming language syntax, semantics, and the runtime environment
2. To be familiarized with general computer programming concepts like conditional execution, loops & functions
3. To learn about mutable and immutable types.
4. To learn about the data science related functions in NUMPY.
5. To solve data science problems using PANDAS.

Course Outcomes: After successful completion of the course, Student will be able to

CO 1	Demonstrate various operators, data types and decision structures in python. (BL - 3)
CO 2	Solve problems using Functions and data structures in Python (BL - 3)
CO 3	Implement the concept of Files and Modules (BL - 3)
CO 4	Implement Data Science queries using NUMPY module (BL - 3)
CO 5	Solve data manipulation task using PANDAS module (BL - 3)

CO-PO Mapping

CO	PO												PSO	
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
CO1	3	1											1	
CO2	2	2											2	
CO3	2	1											2	
CO4	2	2											1	
CO5	2	2											1	1

1: Low, 2-Medium, 3- High

COURSE CONTENT		
MODULE – 1	**I/O and Decision Structures**	**10H**
Input and Output: Introduction to Python and installation, Input and Output, Comments, Variables, Operators. Type conversions, Expressions, Data types. **Decision Structures and Boolean Logic**: if, if-else, if-elif-else Statements, Nested Decision Structures. Looping: while loop, for loop, Nested Loops.		
At the end of the Module 1, students will be able to: 1. Describe python expressions, data types (BL-2) 2. Perform various Arithmetic calculations using Operators in Python(BL-3) 3. Demonstrate the usage of looping structures in python Language.(BL-3)		
MODULE -2	**Functions and Data structures**	10H
Functions: Definition, Function Arguments, Anonymous Function, Scope of the variable and name spacing, Recursion, Map, Filter and Reduce Functions **Strings**, **Lists, Tuples and Dictionaries**: String Methods and Operations, Lists: Operations and Methods, Tuples: Operations and Methods, Dictionaries: Operations and Methods.		
At the end of the Module 2, students will be able to: 1. Implement Functions to solve problems.(BL-3) 2. Describe various String handling functions in python(BL-2) 3. Describe the various **Lists, Tuples and Dictionaries** in python(BL-2)		
MODULE-3	**Files and Modules**	10H
Files: Text Files, File Operations, File Functions, Copying the Files, Two Files Merging into Single File. **Modules:** Modules, Standard Modules, Packages.		
At the end of the Module 3, students will be able to: 1. Describe the concepts of Files (BL-2). 2. Describe the importance of Modules and packages (BL-2).		
MODULE-4	**Introduction to Numpy**	9H
Introduction to Numpy: Fixed-Type Arrays in Python, Creating Arrays from Lists, Creating Arrays from Scratch Numpy Standard Data Types, The Basics of Numpy Arrays, Numpy Array Attributes. **Array Indexing**: Accessing Single Elements, Array Slicing: Accessing Subarrays, Reshaping of Arrays, Array Concatenation and Splitting. Computation on Numpy Arrays: Universal Functions.		
At the end of the Module 4, students will be able to: 1. Describe the concept of Numpy Module(BL-2) 2. Solve numerical problems related to data science using Numpy Arrays.(BL-3) 3. Apply Universal **Functions** for Data Science problems(BL-3)		
MODULE-5	**Data Manipulation with Pandas**	**9H**
Data Manipulation with Pandas: Installing and Using Pandas, Introducing Pandas Objects, Pandas Series Object, Pandas DataFrame Object, Pandas Index Object, Data Indexing and Selection Data Selection in Series. **Data Selection in DataFrame Operating on Data in Pandas Ufuncs**: Index Preservation UFuncs: Index Alignment, Operations Between DataFrame and Series, Handling Missing Data, Trade-Offs in Missing Data Conventions, Missing Data in Pandas, Operating on Null Values		

At the end of the Module 5, students will be able to:

1. Describe the concept of Data Manipulation (BL-2).
2. Describe the concept of Pandas for Data Science(BL-2)
3. Apply Ufunctions in pandas to generate Data Frame (BL-3)

Implement Pandas Module to handle **Missing Data**(BL-3)

Total hours:	**48 HOURS**

Content Beyond Syllabus:

1. Regular Expressions
2. Matplotlib

Text Books:

1. Fundamentals of Python First Programs, Kenneth. A. Lambert, Cengage.
2. Python Data Science Hand Book, Jake Vanderplas, First Edition, Oreilly

REFERENCE BOOK(S):

1. Introduction to Python Programming, Gowrishankar. S, Veena A, CRC Press.
2. Python Programming: A Modern Approach, Vamsi Kurama, Pearson.
3. Python for Data Analysis-Wes McKinney, 2nd-Edition, Oreilly.
4. Python Programming: A Modern Approach, Vamsi Kurama, Pearson. Braun W. J., Murdoch D. J., A First Course in Statistical Programming with R, Cambridge University Press, 2007

NARAYANA ENGINEERING COLLEGE:GUDUR								
21ES1508	**Python Programming and Data Science Lab**							R21
Semester	**Hours / Week**			**Total hrs**	**Credit**	**Max Marks**		
	L	**T**	**P**		**C**	**CIE**	**SEE**	**TOTAL**
II	0	0	2	32	1.5	40	60	100

Pre-requisite: Programming Knowledge

Course Objectives:

1. To gain knowledge on python program basics
2. To prepare students for building programs using control statements
3. To prepare students for solving the problems involving functions and files.
4. To gain knowledge Python Numpy module to solve complex mathematical problems involving matrices.
5. To gain Knowledge of data cleaning using Pandas.

Course Outcomes: After successful completion of the course, the student will be able to:

CO1	Understanding and use of python- Basic Concepts**(BL -2)**
CO2	Solve the problems using python Iterative Statements**(BL -3)**
CO3	Understand the concepts of files, modules**(BL -2)**
CO4	Solve the Numerical problems that involve Matrices **(BL -3)**
CO5	Provide solutions for data cleaning tasks**(BL-3)**

CO-PO Mapping

CO	PO												PSO	
	PO1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	1	1	2										1	
CO2	2	3	2	2									2	1
CO3	2	2	3	2	2								3	2
CO4	2	2	2	1	1								3	2

1-Low, 2-Medium, 3- High

COURSE CONTENT		CO
Task-1	**Python Basics**	**4H**
1. Running instructions in Interactive interpreter and a Python Script 2. Write a program to purposefully raise Indentation Error and Correct it 3. Write a program to compute distance between two points taking input from the user (Pythagorean Theorem) 4. Write a program to convert a Binary number to Decimal number and verify if it is a Perfect number.		CO 1
Task-2	**Conditional Statements**	**2H**
1. Write a program to determine if a given string is a Palindrome or not 2. Write a program for Fibonacci sequence is generated by adding the previous two terms by starting with 1 and 2, the first 10 terms will be: 1, 2, 3, 5, 8, 13, 21, 34, 55, 89,		CO 1
TASK-3	**Functions**	**2H**
1. Write a function that draws a Pyramid with # symbols  2. Choose any five built-in string functions of C language. Implement them on your own in Python. You should not use string related Python built-in functions.		CO2
TASK-4	**Strings**	**4H**
1. Write a program to use split and join methods in the string and trace a birthday with Dictionary data structure. 2. Write a program using map, filter and reduce functions		CO2
TASK-5	**Lists**	**2H**
1. Write program which performs the following operations on lists. Don't use built-in functions a) Updating elements of a list b) Concatenation of list's c) Check for member in the list d) Insert into the list e) Sum the elements of the list f) Push and pop element of list g) Sorting of list h) Finding biggest and smallest elements in the list Finding common elements in the list		CO2
TASK-6	**Files**	**4H**
1. Write a program to read the file content and count the number of vowels, consonants, digits and special characters in a given file. 2. Write a program to perform the following operations in Files: a. Copy from one file to another file Merge two files		CO3
TASK-7	**Introduction to Numpy**	**4H**
1. Write a NumPy program to compute the outer product of two given vectors. Write a Numpy program to compute the determinant of a given square array.		CO4
TASK-8	**Introduction to Numpy**	**2H**
Write a Numpy program to calculate the difference between the maximum andthe minimum values of a given array along the second axis. Expected Output: Original array: [[0 1 2 3 4 5] [6 7 8 9 10 11]] Difference between the maximum and the minimum values of the said array:[5 5]		CO4
TASK-9	**Introduction to Pandas**	**4H**

1. Write a Pandas program to convert a Panda module Series to Python list and it's type. 2. Write a Pandas program to display most frequent value in a given series and replace everything else as 'Other' in the series		CO5
TASK-10	**Introduction to Pandas**	**4H**
1. Write a Pandas program to identify the column(s) of a given DataFrame which have at least one missing value. 2. Write a Pandas program to replace NaNs with a single constant value in specified columns in a DataFrame.		CO5

ADDITIONAL EXPERIMENTS	
TASK – 11 – Lists, Strings, Tuples	
1. Write a python programs on lists 2. Write a python program on strings 3. Write a python program on tuples	CO2
TASK – 12 - Pandas	
1. Write a Pandas program to interpolate the missing values using the Linear Interpolation method in a given DataFrame. 2. Write a Pandas program to import excel data (coalpublic2013.xlsx) into a Pandas DataFrame.	CO5

Virtual Labs	
Python Lab (IIT Bombay) : 1. http://vlabs.iitb.ac.in/vlabs-dev/labs/python-basics/experimentlist.html 2. https://pythoninstitute.org/free-python-courses/?gclid=EAIaIQobChMI4u7Uw-mZ8wIVTR0rCh0CYw2FEAAYAiAAEgL5GPD_BwE	
List of Experiments	
1. Arithmetic Operations 2. Built-in Functions 3. Loops 4. Data Types 5. Strings	6. Classes and Objects 7. Built-in Modules 8. Constructors and Inheritance 9. Numpy basics. 10. Pandas

Text Book(s):

1. Python Programming: A Modern Approach, Vamsi Kurama, Pearson, 2017
2. Learning Python, Mark Lutz, Orielly, 5th Edition, 2013

Reference Book(s):

1. Think Python, Allen Downey, Green Tea Press, 2nd Edition
2. Core Python Programming, W.Chun, Pearson, 2nd Edition, 2007
3. Fundamentals of Python, Kenneth A. Lambert, Cengage Learning, 1st Edition, 2015
4. R. Nageswara Rao, "Core Python Programming", 2nd edition, Dreamtech Press, 2019
5. Allen B. Downey, "Think Python", 2ndEdition, SPD/O'Reilly, 2016
6. Martin C.Brown, "The Complete Reference: Python", McGraw-Hill, 2018.
7. Michael Dawson, ―Python Programming for absolute beginners, 3rd Edition, CENGAGE Learning Publications, 2018.
8. Taming Python by Programming, Jeeva Jose,Khanna Publishing House, 1st Edition, 2018
9. Introduction to Computing and Problem Solving with Python, J. Jose, Khanna Publications,1st Edition, 2019.
10. Guido Van Rossum and Fred L. Drake Jr, "An Introduction to Python – Revised and updated for Python 3.2, Network Theory Ltd., 2011.

NARAYANA ENGINEERING COLLEGE::GUDUR								
21ES1505	IT WORKSHOP							R21
Semester	Hours / Week			Total hrs	Credits	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
II	0	0	3	32	1.5	40	60	100

Pre-requisite:

Course Objectives:

1. To know about the internal parts of a computer, assembling a computer from the parts, preparing a computer for use by installing the operating system
2. To gain knowledge about the usage of tools like Word processors, Spreadsheets, Presentations.
3. To learn about Networking of computers and use Internet facility for Browsing and Searching

CourseOutcomes:Aftersuccessfulcompletionofthecourse,thestudentwillbeableto:

CO1	Build a Personal Computer and prepare the computer ready to use(BL-2)
CO2	Apply knowledge to Interconnect two or more computers for information sharing (BL-3)
CO3	Prepare documentation for projects and other assignments (BL-3)
CO4	Demonstrate seminars and other assignments using presentation tools (BL-3)
CO5	Analyze data using spread sheets (BL-3)

CO-PO Mapping

	PO												PSO	
CO	**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	2				3								1	3
CO2	2				3								1	3
CO3	2				3								1	3
CO4	2				3								1	3
CO5	2				3								1	3

1: Low, 2-Medium, 3- High

COURSE CONTENT		CO
Task-1	**Learn about Computer**	**3H**
Identify the internal parts of a computer, and its peripherals. Represent the same in the form of diagrams including Block diagram of a computer. Write specifications for each part of a computer including peripherals and specification of Desktop computer. Submit it in the form of a report		CO1
Task-2	**Assembling a Computer**	**3H**
Disassemble and assemble the PC back to wor2k5ing condition. Students should beable to trouble shoot the computer and identify working and non-working parts. Student should identify the problem correctly by various methods		CO1
TASK-3	**Install Operating system**	**3H**
Student should install Linux on the computer. Student may install another operating system (including proprietary software) and make the system dual boot or multi boot. Students should record the entire installation process.		CO1
TASK-4	**Operating system features**	**3H**
Students should record the various features that are supported by the operating system(s) installed. They have to submit a report on it. Students should be able to access CD/DVD drives, write CD/DVDs, access pen drives, print files, etc. Students should install new application software and record the installation process.		CO1
TASK-5	**Networking**	**3H**
Students should connect two computers directly using a cable or wireless connectivity and share information. Students should connect two or more computersusing switch/hub and share information. Crimpling activity, logical configuration etc. should be done by the student. The entire process has to be documented.		CO2
TASK-6	**Browsing Internet**	**3H**
Student should access the Internet for Browsing. Students should search the Internet for required information. Students should be able to create e-mail account and send email. They should get acquaintance with applications like Face book, Skype etc. If Intranet mailing facility is available in the organization, then students should share the information using it. If the operating system supports sending messages tomultiple users (LINUX supports it) in the same network, then it should be done by the student. Students are expected to submit the information about different browsers available, their features, and search process using different natural languages, and creating email account.		CO2
TASK-7	**Antivirus**	**3H**
Students should download freely available Antivirus software, install it and use it to check for threats to the computer being used. Students should submit information about the features of the antivirus used, installation process, about virus definitions, virus engine etc.		CO2
TASK-8	**Word Processor**	**3H**
Students should be able to create documents using the word processor tool. Some of the tasks that are to be performed are inserting and deleting the characters, words and lines, Alignment of the lines, Inserting header and Footer, changing the font, changing the colour, including images and tables in the word file, making page setup, copy and paste block of text, images, tables, linking the images which are present in other directory, formatting paragraphs, spell checking, etc. Students should be able to prepare project cover pages, content sheet and chapter pages at the end of the task using the features studied. Students should submit a user manual of the word processor considered, Image Manipulation tools.		CO3
TASK-9	**Presentations**	**3H**

creating, opening, saving and running the presentations, selecting the style for slides, formatting the slides with different fonts, colours, creating charts and tables, inserting and deleting text, graphics and animations, bulleting and numbering, hyper linking, running the slide show, setting the timing for slide show.		CO4
TASK-10	**Spreadsheet**	**3H**
Students should be able to create, open, save the application documents and format them as per the requirement. Some of the tasks that may be practiced are Managing the worksheet environment, creating cell data, inserting and deleting cell data, format cells, adjust the cell size, applying formulas and functions, preparing charts, sorting cells. Students should submit a user manual of the Spreadsheet		CO5
Additional Experiments		
TASK-1	**LateX**	**2H**
Introduction to Latex and its installation and different IDEs. Creating first document using Latex, using content into sections using article and book class of LaTeX . Styling Pages: reviewing and customizing different paper sizes and formats. Formatting text (styles, size, alignment, colors and adding bullets and numbered items, inserting mathematical symbols, and images, etc.). Creating basic tables, adding simple and dashed borders, merging rows and columns. Referencing and Indexing: cross-referencing (refer to sections, table, images), bibliography (references).		CO5

Text Book(s):

1. B.Govindarajulu, "IBM PC and Clones Hardware Trouble shooting and Maintenance",2nd edition, Tata McGraw-Hill, 2002
2. "MOS study guide for word, Excel, Powerpoint& Outlook Exams", Joan Lambert, Joyce Cox, PHI.
3. "Introduction to Information Technology", ITL Education Solutions limited, Pearson Education.

Reference Book(s):

1. Rusen, "Networking your computers and devices", PHI
2. Bigelows, "Trouble shooting, Maintaining & Repairing PCs", TMH.

SEMESTER-III

NARAYANA ENGINEERING COLLEGE::GUDUR								
21ES1009	**DATA STRUCTURES AND ALGORITHMS**							**R21**
Semester	**Hours / Week**			**Total hrs**	**Credit**	**Max Marks**		
	L	**T**	**P**		**C**	**CIE**	**SEE**	**TOTAL**
III	3	0	0	48	3	40	60	100

Pre-requisite: Knowledge of Mathematics, Computer Programming, Analytical & Logical Skills

Course Objectives:

1. To explain efficient storage mechanisms of data for an easy access.
2. To design and implementation of various basic and advanced data structures.
3. To introduce various techniques for representation of the data in the real world.
4. To develop applications using data structures.
5. To pertain knowledge on improving the efficiency of algorithm by using suitable data structure.

Course Outcomes: After successful completion of the course, student will be able to:

CO 1	Analyze the data structure algorithms to evaluate the time & space complexities. (BL-4)
CO 2	Apply the knowledge of stack and queues for various applications. (BL - 3)
CO 3	Construct the linked lists for various applications. (BL - 3)
CO 4	Apply the knowledge of tree data structures for various applications. (BL - 3)
CO 5	Develop the graph models of the given problem through graph concepts(BL - 3)

CO-PO Mapping

CO	**PO**												**PSO**	
	PO1	**PO2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO1**	**PSO 2**
CO 1	3	3	2										2	3
CO 2	3	3	3	2									2	2
CO 3	1	2	3	3									2	2
CO 4	2	2	2	2									2	2
CO 5	2	1	3	1									3	2
1: Low, 2-Medium, 3- High														

COURSE CONTENT		
MODULE – 1	**Introduction to Data Structures**	**9H**
Introduction: Overview of Data Structures, Implementation of Data Structures, Algorithm Specifications, Analysis of an Algorithm, Asymptotic Notations, Time-Space trade off, Arrays. **Searching:** Introduction, Basic Terminology, Linear Search and Binary Search Techniques and their complexities.		
At the end of the Module 1, students will be able to: 1. Understand the linear and non-linear data structures. (BL - 2) 2. Understand the time and space complexities of an algorithm. (BL - 2) 3. Illustrate representation of data using Arrays. (BL - 2) 4. Explain searching techniques. (BL - 2)		
MODULE -2	**Stacks and Queues**	**9H**
Stacks: Introduction, Representation of a Stack, Stack Operations, Applications of Stacks. **Queues:** Introduction, Representation of a Queue, Queue Operations, Various Queue Structures: Circular Queue, Double Ended Queue, Priority Queue, Applications of Queues.		
At the end of the Module 2, students will be able to: 1. Explain stack ADT and its operations. (BL - 2) 2. Understand the expression evaluation using stacks. (BL - 2) 3. Implement various queue structures. (BL - 3)		
MODULE-3	**Linked Lists and Sorting**	**10H**
Introduction, Singly linked lists, Doubly Linked Lists, Circular Linked Lists, Linked Stacks and Queues, Applications of Linked Lists. **Sorting:** Introduction, Bubble Sort, Selection Sort, Insertion Sort, Merge Sort, Quick Sort		
At the end of the Module 3, students will be able to: **1.** Understand basics concepts of linked lists. **(BL - 2)** **2.** Illustrate various structures of linked lists. **(BL - 2)** **3.** Understand the concept of sorting. **(BL - 2)**		
MODULE-4	**Trees**	**10H**
Introduction, Basic Terminologies, Definition and concepts, Representation of Binary Tree, operations on a Binary Tree, Binary Search Tree, Height balanced Binary Tree, B Trees.		
At the end of the Module 4, students will be able to: **1.** Understand the concept of trees. **(BL - 2)** **2.** Compare different tree structures. **(BL - 2)** **3.** Apply trees for indexing. **(BL - 3)**		
MODULE-5	**Graphs & Hashing**	**10H**
Graphs: Introduction, Graph Terminologies, Representation of Graphs, Graph Operations, Shortest Paths, Topological Sorting, Minimum Spanning Trees – Kruskal's and Prim's algorithms. **Hashing:** Introduction to Hash Table, Static Hashing, Dynamic Hashing.		
At the end of the Module 5, students will be able to: **1.** Explain the importance of Graphs for solving problems. **(BL - 2)** **2.** Understand graph traversal methods. **(BL - 2)** Implement algorithms to identify shortest path. **(BL - 3)**		
	Total hours	**48 hours**

Content beyond syllabus:

- Activation Record Management
- Optimum Sorting Algorithms

TEXT BOOK(S):

3. D. Samanta, **Classic Data Structures**, 2nd Edition, Prentice-Hall of India, Pvt. Ltd.,India, 2012.
4. Ellis Horowitz and SartajSahni, **Fundamentals of Data Structures in C**, 2nd Edition, Universities Press, 2008.

REFERENCE BOOK(S):

1. Data Structures A Pseudo code Approach with C, Second Edition by Richard F.Gilberg, Behrouz A. Forouzan, Cengage Learning.
2. Data Structures and Algorithms Using C++ by Ananda Rao Akepogu, RadhikaRaju Palagiri, Pearson, 2010.
3. Data Structures and Algorithms Made Easy by Narasimha Karumanchi, CareermonkPublications, 2016
4. Peter Bras, "Advanced Data Structures", Cambridge University Press, 2014
5. Data Structures, RS Salaria, Khanna Publishing House, 3rd Edition, 2017
6. Data Structures through C, Yashwant Kanetkar, BPB Publications, 3rd Edition, 2019

Expert Data Structures with C, RB Patel, Khanna Publications, 2019

NARAYANA ENGINEERING COLLEGE::GUDUR								
21CS2001	**COMPUTER ORGANIZATION & ARCHITECTURE**							**R21**
Semester	**Hours / Week**			**Total hrs**	**Credit**	**Max Marks**		
	L	**T**	**P**		**C**	**CIE**	**SEE**	**TOTAL**
III	3	0	0	48	3	40	60	100

Pre-requisite: Computer fundamentals and Digital Logic Design.

Course Objectives:

1. To learn the fundamentals of computer organization and its relevance to classical and modern problems of computer design.
2. To understand the structure and behavior of various functional modules of a computer.
3. To design logical expressions and corresponding integrated logic circuits for a variety of problems.
4. To understand the internal organization and operations of a computer.
5. To introduce the concepts of processor logic design and control logic design.

Course Outcomes: After successful completion of the course, the student will be able to:

CO1	Describe the concepts of Functional Architecture and Basic Operations of Computing System. (BL-2)
CO2	Interpret there presentation of Fixed and Floating point numbers stored in digital computer. (BL-3)
CO3	Illustrate the basics of Instruction set and design of control units to execute Computer instruction. (BL - 3)
CO4	Analyze the Memory System and their impact on Computer cost & performance. (BL - 4)
CO5	Demonstrate the basic knowledge of I/O devices and Interfacing of I/O devices with computer.(BL - 3)

CO-PO Mapping														
CO	**PO**												**PSO**	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	3	3											3	2
CO2	2	3											3	3
CO3	2	3											3	3
CO4	3	2											2	2
CO5	3	3											3	3
1: Low, 2-Medium, 3- High														

COURSE CONTENT		
MODULE – 1	**Introduction of computer architecture**	**10H**
Basic Structure of Computer: Computer Types, Functional Units, Basic operational Concepts, Bus Structure, Performance, Multiprocessors and Multicomputer. **Machine Instructions and Programs:** Numbers, Arithmetic Operations and Programs, Instructions and Instruction Sequencing, Addressing Modes, Basic Input/output Operations, Stacks and Queues, Subroutines.		
At the end of the Module 1, students will be able to: 1. Illustrate the basic functional units and different ways of interconnecting to form a computer system. (BL 2). 2. Compare Multiprocessors and Multicomputer. (BL 2). 3. Explain addressing modes for accessing register and memory operands.(BL 2). 4. Define Input/output Operations. (BL 1).		
MODULE – 2	**Data representation and computer Arithmetic**	**9H**
Fixed point representation of numbers: Algorithms for arithmetic operations, multiplication (Booths, Modified Booths), division (restoring and non-restoring). **Floating point representation:** IEEE standards and algorithms for common arithmetic operations- Representation of non-numeric data (character codes).		
At the end of the Module 2, students will be able to: 1. Explain fixed point and floating point representation of numbers. (BL 2). 2. Make use of IEEE standards to perform operations on floating point numbers. (BL 3). 3. Apply Booths algorithm to multiply two signed numbers. (BL 3).		
MODULE-3	**Concepts of Computer Architecture**	**9H**
Introduction to ISA (Instruction Set Architecture): Machine Instruction Characteristics, Types of operands, Instruction formats, Instruction types and addressing modes. **Basic Processing Unit:** Fundamental Concepts, Execution of a Complete Instruction, Multiple Bus Organization, Hardwired Control, Micro programmed Control.		
At the end of the Module 3, students will be able to: 1. Discuss the Machine Instruction Characteristics. (BL 2). 2. Explain Instruction types and addressing modes. (BL 2). 3. Define the concept of Multiple Bus Organization (BL 1). 4. Compare hardwired and micro programmed control units. (BL 2).		
MODULE-4	**Memory Organization**	**10H**
Memory System: Basic concepts, Semiconductor RAM memories, Read only memories, speed, size and cost, Cache memories, performance considerations, Virtual memory, Memory management requirements, Secondary storage. **Large Computer Systems:** Forms of Parallel Processing, Array Processors, The Structure of General-Purpose multiprocessors, Interconnection Networks, Data Hazards, Instruction Hazards.		
At the end of the Module 4, students will be able to: 1. Recognize the various types of memories. (BL 1). 2. Understand the concept of memory organization. (BL 2). 3. Explain the concept of Multiple Bus Organization. (BL 2). 4. Compare the performance of cache memory and virtual memory. (BL 2). Understand the Interconnection Networks structure and hazards of the system (BL2).		

MODULE-5	Input/Output Organization	10H
I/O Basics: Accessing I/O Devices, Interrupts: Interrupt Hardware, Enabling and Disabling Interrupts, Handling Multiple Devices, Direct Memory Access (DMA). **Buses:** Synchronous Bus, Asynchronous Bus, Interface Circuits, Standard I/O Interface, Peripheral Component Interconnect (PCI) Bus, Universal Serial Bus (USB).		
At the end of the Module 5, students will be able to: 1. Understand I/O Devices and buses. (BL 2). 2. Make use of interrupt handling mechanisms for various processors. (BL 3). 3. Describe the concept of DMA. (BL 2). Understand Interface Circuits and Standard I/O Interface. (BL 2).		
	Total Hours	**48H**

Content beyond syllabus:

1. Signed magnitude numbers addition on various numbers.
2. PLA control.

Text Book(s):

1. Carl Hamacher, ZvonkoVranesic, SafwatZaky, “Computer Organization”, 5th Edition, McGraw Hill Education, 2013.
2. David A. Patterson and John L. Hennessy Computer Organization and Design-The Hardware/Software Interface 5th edition, Morgan Kaufmann, 2013.

Reference Book(s):

1. Mano M. M., Digital Logic & Computer Design, 4/e, Pearson Education, 2013.
2. W. Stallings, Computer organization and architecture, 8th edition, Prentice-Hall, 2013.
3. Patterson D.A. and J. L. Hennessey, Computer Organization and Design, 5/e, Morgan Kauffmann Publishers, 2013.
4. William Stallings, Computer Organization and Architecture: Designing for Performance, 9/e, Pearson, 2013.
5. Chaudhuri P., Computer Organization and Design, 2/e, Prentice Hall, 2008.

NARAYANA ENGINEERING COLLEGE::GUDUR								
21CS2002	**DATABASE MANAGEMENT SYSTEMS**							**R21**
Semester	**Hours / Week**			**Total hrs**	**Credit**	**Max Marks**		
	L	**T**	**P**		**C**	**CIE**	**SEE**	**TOTAL**
III	3	0	0	48	3	40	60	100

Pre-requisite: Knowledge of File Structures, Data Structures

Course Objectives:

1. To teach the role of database management system in an organization.
2. To design databases using data modeling and Logical database design techniques.
3. To construct database queries using relational algebra and calculus and SQL.
4. To explore implementation issues in database transaction.
5. To familiarize database security mechanisms.

Course Outcomes: On successful completion of the course, the student will be able to:

CO 1	Describe database technologies and database design. (BL-2)
CO 2	Understand Relational Database Management Systems. (BL-2)
CO 3	Construct queries, procedures for database creation in RDBMS.(BL-3)
CO 4	Apply normalization on database design. (BL-3)
CO 5	Demonstrate concurrency control techniques and techniques for database recovery. (BL-2)

CO-PO Mapping

CO	PO												PSO	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	3	3	3										3	3
CO2	3	3	2		3								3	2
CO3	3	2	2		2								2	3
CO4	3	2	3		3								2	3
CO5	2	3	3										3	2

1: Low, 2-Medium, 3- High

COURSE CONTENT		
MODULE – 1	**Introduction to Database concepts and Modeling**	**8H**
Conceptual Modelling Introduction: Introduction to Data bases, Purpose of Database Systems, View of Data, Data Models, Database Languages, Database Users, Database Systems architecture. **The Entity-Relationship Model:** Overview of Database Design, Beyond ER Design, Entities, Attributes and Entity sets, Relationships and Relationship sets, Conceptual Design with the ER Model.		
At the end of the Module 1, students will be able to: 1. Understand the Purpose of Database Systems, Data Models, and View of Data.(BL-2) 2. Summarize the concept of Database Languages, Users and Architecture. (BL-2) 3. Design ER diagrams for given database. (BL-2) 4. Explain conceptual design for enterprise systems (BL-2)		
MODULE – 2	**Relational Model, Relational Algebra**	**8H**
Relational Model: Introduction to the Relational Model – Integrity Constraints over Relations, Enforcing Integrity constraints, querying relational data, Logical data base Design, Views. **Relational Algebra:** Introduction to Relational algebra, selection and projection, set operations, renaming, joins, division.		
At the end of the Module 2, students will be able to: 4. Understand Basics of Relational Model. (BL-2) 5. Describe phases of Logical Database Design.(BL-2) **6.** Explain the relational algebra operations on relations. (BL-2)		
MODULE – 3	**SQL**	**8H**
SQL: Basic form of SQL Query, DDL, DML, Views in SQL, Joins, Nested & Correlated queries, Operators, Aggregate Functions, integrity and security, Functions & Procedures, Packages, Triggers, Cursors, PL/SQL principles and examples.		
At the end of the Module 3, students will be able to: 1. Construct SQL queries in RDBMS. (BL-3) 2. Understand integrity and security Constraints in SQL (BL-2) 3. Construct PL/SQL programs in RDBMS. (BL-3)		
MODULE – 4	**Normalization & Transaction Management**	**12H**
Relational database design: Introduction, Functional Dependencies (FDs), Normalization for relational databases: 1NF, 2NF, 3NF and BCNF, Basic definitions of Multi Valued Dependencies, 4NF and 5NF. **Transaction Management:** Transaction processing, Transaction Concept, Transaction State, Implementation of Atomicity and Durability, Concurrent Executions.		
At the end of the Module 4, students will be able to: 1. Analyze functional dependencies. (BL-3) 2. Apply normal forms on functional dependencies. (BL-3) 3. Understand Atomicity and Durability, Concurrent Executions. (BL-2)		
MODULE – 5	**Concurrency Control & Recovery and Indexing**	**12H**
Concurrency Control: Lock-Based Protocols, Timestamp- Based Protocols, Validation-Based Protocols, Multiple Granularity. **Recovery:** Failure Classification, Recovery and Atomicity, Log-Based Recovery. **Indexing:** Introduction to Index data structures, Hash-Based, Tree Based Indexing.		
At the end of the Module 5, students will be able to: 1. Discuss the Concurrency Control and various Protocols. (BL-2)		

2. Understand reasons for system failures. (BL-2) 3. Understand Ordered Indices, B+ Tree Index Files. (BL-2)	
Total hours	**48 Hours**

Content beyond syllabus:

1. Embedded SQL
2. Client/Server Database environment
3. Web Database environment

Text Book(s):

1. Abraham Silberschatz, Henry F. Korth, S. Sudarshan, Database System Concepts, 6th Edition, Tata McGraw-Hill Publishing Company,2017.
2. Raghu Ramakrishnan, Database Management System, 3rd Edition, Tata McGraw-Hill Publishing Company, 2014.

Reference Book(s):

1. Peter Rob, A.Ananda Rao, Corlos Coronel, Database Management Systems (for JNTU), Cengage Learning, 2011.
2. Hector Garcia Molina, Jeffrey D. Ullman, Jennifer Widom, Database System Implementation, 1st Edition, Pearson Education, United States, 2000.
3. E. Ramez and Navathe, Fundamental of Database Systems, 7th Edition, Pearson Education
4. R.P. Mahapatra & Govind Verma, Database Management Systems, Khanna Publishing House, 2016.
5. Carlos Coronel and Steven Morris, Database Systems: Design, Implementation, and Management, 12th edition, Cengage Learning,2016.
6. John V. ,Absolute beginner's guide to databases, Petersen, QUE

NARAYANA ENGINEERING COLLEGE::GUDUR								
21CS2003	**MATHEMATICAL FOUNDATION FOR COMPUTER SCIENCE**							**R21**
Semester	**Hours / Week**			**Total hrs**	**Credit**	**Max Marks**		
	L	**T**	**P**		**C**	**CIE**	**SEE**	**TOTAL**
III	3	0	0	60	3	40	60	100

Pre-requisite: Student need to have knowledge in mathematical basics in computers

Course Objectives:

- To covert the statements logical expressions and logical theorem proving.
- Understand the basics to design the hasse diagrams.
- Understand the homomorphism and Isomorphism concepts by algebraic structures.
- To understand the basics of counting methods.
- Understanding the recurrence relations and generating functions by mathematical induction.
- To understand of basics of trees and graphs.

Course Outcomes: After successful completion of the course, the student will be able to:

CO 1	Understand the concepts associated with Mathematical Logic and Predicate calculus
CO 2	Learn The Basic Concepts About Relations, Functions, Algebraic Structures And To Draw Different Diagrams Like Lattice, Hasse Diagrams
CO 3	Understand The Elementary Combinatory And Pigeon-Hole Principle.
CO 4	Describe Functions, Various Types Of Recurrence Relations And The Methods To Find Out Their Solutions.
CO 5	Understand The Basic Concepts Associated With Graphs And Trees

CO-PO Mapping														
CO	**PO**												**PSO**	
	PO1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	**2**	**2**											**2**	
CO2	**2**	**3**	**1**											
CO3	**3**	**3**												
CO4	**3**	**3**	**2**											
CO5	**3**	**1**	**3**											
1: Low, 2-Medium, 3- High														

COURSE CONTENT		
MODULE – I	**STATEMENTS AND PREDICATE CALCULUS**	**10 Hrs**
Statements and notations, connectives, well-formed formulas, truth tables, tautology, Equivalence implication; Normal forms: Disjunctive normal forms, Conjunctive normal forms, Principle Disjunctive normal forms, Principle Conjunctive normal forms.Predicative logic, Free & Bound variables, Rules of inference, Consistency, proof of contradiction		
At the end of this Module students will be able: 1. To understand the concepts associated with Mathematical Logic and Predicate calculus.		
MODULE- II	**SET THEORY**	**11Hrs**
Properties of binary relations, equivalence, compatibility and partial ordering relations, lattices, Hasse diagram. Inverse function, composition of functions, recursive functions. Lattices as partially ordered sets; Definition and examples, properties of lattices.Algebraic systems, Examples and general properties, Semi groups and Monoids, groups, and sub groups, homomorphism, Isomorphism.		
At the end of this Module students will be able: 1. To learn the basic concepts about relations, functions and to draw different diagrams like Lattice, Hasse diagrams. 2. To understand the concepts of Algebraic Structures and combinatorics.		
MODULE- III	**ELEMENTARY COMBINATORICS**	**9 Hrs**
Basics of counting, Permutations and Combinations, permutations and combinations with repetitions, the binomial theorem, multinomial theorem, generalized Inclusion-Exclusion principle, Pigeon-hole principle and its applications.		
At the end of this Module students will be able: **1.** To understand the Elementary Combinatorics and Pigeon-hole principle.		
MODULE- IV	**GENERATING FUNCTIONS &RECURRENCE RELATIONS**	**9 Hrs**
Function of Sequences, Calculating Coefficients of generating functions.Recurrence relations, Solving recurrence relation by substitution and Generating functions, the method of Characteristic roots, solution of Inhomogeneous Recurrence Relations.		
At the end of this Module students will be able: 1. To describe various types of recurrence relations and the methods to find out their solutions.		
MODULE- V	**GRAPH THEORY**	**10 Hrs**
Basic concepts of graphs, isomorphic graphs, Euler graphs, Hamiltonian graphs, planar graphs, graph coloring, digraphs, directed acyclic graphs, weighted graphs, Chromatic numbers. Trees, BFS, DFS, Spanning trees, Minimal spanning trees.		
At the end of this Module students will be able: 1. To understand the basic concepts associated with Graphs and Trees.		
	Total hours:	**49 Hours**

Content beyond syllabus:

Finding Minimal cost Spanning Tree using Prim's Algorithm.

Text Book(s):

1. Discrete Mathematical Structures with Applications to Computer Science, J.P.Tremblay, R.Manohar, Mc.Grahill, 2001.
2. Discrete Mathematics and its Applications, Kenneth H.Rosen, 6th edition, TMH.
3. Mathematical Foundations of Computer Science, P.Chandrasekharaiah, Prism publications.

Reference Book(s):

1. Discrete Mathematics for Computer Scientists & Mathematicians, second edition, J.L.Mott, A. Kandel, T.P. Baker, PHI
2. Discrete Mathematical Structures, Mallik and Sen, Cengage Learning.
3. Discrete Mathematical Structures, BernandKolman, Robert C. Busby, Sharon Cutler Ross, PHI/ Pearson Education.

NARAYANA ENGINEERING COLLEGE::GUDUR								
21CS2004	**OBJECT ORIENTED PROGRAMMING THROUGH JAVA**							R21
Semester	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
III	3	0	0	48	3	40	60	100

Pre-requisite: Basic knowledge of programming.

Course Objectives:

1. To acquire knowledge on preliminaries of Java.
2. To provide sufficient knowledge on developing real world problems.
3. To demonstrate the principles of packages, inheritance and interfaces.
4. To understand exception handling and Multi threading.
5. To understand the concepts of Applets and I/O Files.

Course Outcomes: After successful completion of the course, Student will be able to:

CO1	Describe the basic Elements of Java for problem solving.(BL-2)
CO2	Demonstrate the concepts of arrays and strings for organizing data. (BL-3)
CO3	Describe the concepts of object oriented programming. (BL-2)
CO4	Design the web applications through java applets. .(BL-3)
CO5	Develop Multi-threaded programs to improve the system performance. (BL-6)

CO-PO Mapping

CO	**PO**												**PSO**	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	3	2											3	3
CO2	2	2	2										2	2
CO3	2	3	2										2	3
CO4	2	3	3										3	2
CO5	3	3	3										3	3

1: Low, 2-Medium, 3- High

COURSE CONTENT		
MODULE – 1	**Basic concepts of java**	**9H**
The History and Evolution of java: History of java, The java Buzz words, The Evolution of java, Lexical issues. **Data types, variables:** Data types, Variables, The Scope and Life time of variables, Operators, Expressions, Control statements, Type conversion and casting, Command Line Arguments.		
At the end of the Module 1, students will be able to: 1. Explain the importance of java. (BL-2) 2. Identify various basic components of java. (BL-2) 3. Implement programs on fundamental concepts of java. (BL-3)		
MODULE -2	**Arrays and Strings**	**9H**
Arrays: Declaration, Initialization and accessing values, One-Dimensional Arrays, Multi-dimensional arrays, Alternative Array Declaration Syntax, var-arg methods, Wrapper Classes.		
Strings: String, StringBuffer and StringBuilder classes.		
At the end of the Module 2, students will be able to: 1. Understand Arrays and accessing array values.(BL-2) 2. Demonstrate 1-D and Multi-dimensional arrays.(BL-2) **3.** Explain the String, StringBuffer, StringBuilder Classes.(BL-2)		
MODULE-3	**OOPs Concepts**	**10H**
Introducing classes: Class fundamentals. Declaration objects, Assigning object reference variables, Introducing Methods, Constructors, this keyword, Garbage collection. **Inheritance and Polymorphism:**: Inheritance basics, Types of inheritance, Benefits, Member access rules, Constructor and calling sequence, Abstract Classes, Super and final keywords**.** Method overloading and Method overriding. **Interfaces**: Defining an interface, Extending interfaces, Implementing interface, Accessing interface properties.		
At the end of the Module 3, students will be able to: 1. Understand the basic syntax for class fundamentals.(BL-2) 2. Explain Access modifiers in Inheritance.(BL-2) 3. Compare and Contrast Method overloading and Method overriding.(BL-3) 4. Explain interface and its implementation.(BL-2)		
MODULE-4	**Packages , Exception Handling and Applets**	**10H**
Packages: Defining Package, finding packages and class path, accessing Protection**.** **Exception Handling**: Exception handling Fundamentals, exception types, Built-in Exceptions, Using try-catch-finally throw- throws keywords, creating your own Exceptions. **Applets:** Introduction to Applets, Applet Life Cycle methods.		
At the end of the Module 4, students will be able to: 1. Develop user defined packages.(BL-3) 2. Implement Exception Handling.(BL-3) 3. Write our own Exceptions (BL-1) 4. Implement Applet Life Cycle Methods. (BL-3)		
MODULE-5	**Multi-Threaded Programming and Files**	**10H**
Multi-Threaded Programming: The java thread model, Thread Life Cycle, The main() thread, creating a Thread, Creating Multiple Threads, Using isalive() and join(), Thread Priorities, Synchronization. **I/O Files:** Byte Oriented and Character oriented classes, RandomAccess Files.		

At the end of the Module 5, students will be able to: 5. Explain the concept of multi threaded concept.(BL-2) 6. Discuss thread states and its priorities.(BL-3) 7. Understand the concept of Synchronization.(BL-2) 8. Demonstrate input/output Files.(BL-3)	
Total hours:	**48 Hours**

Content beyond syllabus:

1. Event Handling Mechanism
2. GUI Programming in JAVA

Text Book(s):

1. Herbert Scheldt, "Java The complete reference", 9th edition, McGraw Hill Education (India) Pvt. Ltd.
2. Ivor Horton, Beginning Java 2, JDK 5th Edition, Wiley Dreamtech.

Reference Book(s):

1. R A. Johson-Thomson, An introduction to java programming and object oriented application development,
2. Y Daniel liang, Introduction to java programming 6th Edition, Pearson Education.
3. C. Xavier, Java programming: A practical approach, First edition, TMH, 2011.
4. Bruce Eckel, Thinking in Java, 2nd Edition, Pearson Education
5. H.M Dietel and P.J Dietel, Java How to Program, 6th Edition, Pearson Ed.
6. Y. Daniel Liang, Introduction to Java programming-comprehensive, 10E, Pearson ltd 2015.
7. E Balagurusamy, Programming With Java: A Primer 5th Edition Tata McGraw Hill.

NARAYANA ENGINEERING COLLEGE::GUDUR								
21ES1513	**Data Structures and Algorithms Lab**							**R21**
Semester	**Hours / Week**			**Total hrs**	**Credit**	**Max Marks**		
	L	**T**	**P**		**C**	**CIE**	**SEE**	**TOTAL**
III	0	0	3	48	1.5	40	60	100

Pre-requisite: Knowledge of Mathematics, Computer Programming, Analytical & Logical Skills

Course Objectives:

1. To introduce various data structures.
2. To elucidate how the data structure selection influences the algorithm complexity.
3. To explain the different operations that can be performed on data structures.
4. To introduce to the search and sorting algorithms.

Course Outcomes: After successful completion of the course, Student will be able to:

CO 1	Apply the Arrays and linked lists for solving the problems. (BL -3)
CO 2	Apply the stacks and queues for solving the given applications. (BL -3)
CO 3	Implement operations on binary trees and binary search trees for given applications. (BL -3)
CO 4	Implement searching and sorting algorithms for given applications. (BL -3)

CO-PO Mapping

CO	**PO**												**PSO**	
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1	3	3	3						2	2			2	2
CO2	3	3	3						2	2			2	2
CO3	3	2							2	2			2	2
CO4	3	3	3		2				2	2			2	2
1: Low, 2-Medium, 3- High														

COURSE CONTENT	CO
TASK-1	**(3H)**
1. Write a Program to Implement the following Searching Algorithms: a) Linear Search b) Binary Search	CO1
TASK-2	**(6H)**
1. Implement the following using arrays: A. Write a Program to Implement Stack Operations B. Write a Program to convert a given infix expression into its Postfix using stack. C. Write a Program to evaluate the Postfix Expression using stack	CO1
TASK-3	**(3H)**
1. Write a Program to Implement Queue Operations using Arrays 2. Write a Program to Implement Circular Queue Operations using Arrays	CO2
TASK-4	**(6H)**
1. Write a Program to implement the operations of Singly Linked List **2.** Write a Program to implement the operations of Doubly Linked List	**CO2**
TASK-5	**(6H)**
1. Write a Program to implement stack operations using linked list **2.** Write a Program to implement the operations of Circular Singly Linked List	**CO3**
TASK-6	**(3H)**
1.Write a Program to Sort the set of elements: **a)** Insertion Sort b) Quick Sort	**C04**
TASK-7	**(3H)**
1. Write a Program to Sort the set of elements: **a)** Merge Sort b) Heap Sort	**C04**
TASK-8	**(6H)**
1. Write a Program to implement the following on trees a) Insertion and deletion operations b) Traversals **2.** Write a Program to implement Binary Search Tree Operations.	**CO3**
TASK-9	**(6H)**
1. Write a Program to implement the following Graph Traversal Algorithms: **a)** Depth first traversal b) Breadth first traversal	**CO4**
TASK-10	**(6H)**
1. Write a Program to implement the following Minimum Spanning Tree Algorithms: **a)** Kruskal's Algorithm b) Prim's Algorithm	**CO4**

Additional Experiments:	
TASK-1	
Write Program to Implement Fibonacci Search Write a Program to Implement Double Ended Queue Operations by using Array	CO4
TASK-2	
1. Write a Program to Implement Tree traversal Techniques 2. Write a Program to Implement Radix Sort	CO4

Virtual Labs:

1. Data Structures – 1 (IIIT HYDERABAD) : https://ds1-iiith.vlabs.ac.in/data-structures-1/

List of Experiments

Sorting
1. Bubble Sort
2. Merge Sort
3. Heap Sort
4. Quick Sort

Graphs
1. Depth First Search
2. Breadth First Search

Trees
1. **Tree Traversal**
2. **Binary Search Trees**

Stacks and Queues
1. Stacks and Queues
2. Infix to Postfix

Searching
1. Unsorted Arrays
2. Hashtables

Linked Lists
1. Linked lists
2. Polynomial Arithmetic using linked lists

2. Data Structures – 2 (IIIT HYDERABAD) : https://ds2-iiith.vlabs.ac.in/data-structures-2/

List of Experiments

Sorting
1. Selection Sort
2. Radix Sort

Graphs
1. Topological Sort
2. Minimum Spanning Trees
3. Path algorithms: Dijkstra's shortest path

Search Trees
1. 2-3 Tree
2. Red Black Tree

Strings
1. Tries and Suffix Trees
2. Substring search: KMP algorithm

Text Book(s):

1. D. Samanta, "Classic Data Structures", 2nd Edition, Prentice-Hall of India, Pvt. Ltd., India, 2012.
2. Horowitz Sahni and Anderson-Freed —Fundamentals of Data Structures in C. 2nd Edition,
Universities Press, 2008.

Reference Book(s):

1. Richard F. Gilberg& B. A. Forouzan —Data Structures A Pseudocode Approcah with C, Second Edition, CENGAGE Learning.
2. Ananda Rao,Data Structures and Algorithms Using C++,Akepogu, Radhika Raju Palagiri, Pearson, 2010.
3. Mark Allen Weiss, Data structure and Algorithm Analysis in C. Addison Wesley Publication. 2006.
4. Jean Paul Trembley and Paul G. Sorenson, An Introduction to Data Structures with Applications, 2ndEdition, McGraw Hill Education, 2017
5. Thomas Cormen, C. Leiserson, R. L. Rivest and C. Stein, —Introduction to Algorithms, 2nd Edition, PHI, 2010
6. Narasimha Karumanchi, Data Structures and Algorithms Made Easy, Careermonk Publications, 2016
7. Peter Bras, Advanced Data Structures, Cambridge University Press, 2014
8. Data Structures, RS Salaria, Khanna Publishing House, 3rd Edition, 2017
9. Data Structures through C, Yashwant Kanetkar, BPB Publications, 3rd Edition, 2019
10. Expert Data Structures with C, RB Patel, Khanna Publications, 2019

NARAYANA ENGINEERING COLLEGE::GUDUR								
21CS2501	**DATABASE MANAGEMENT SYSTEMS LAB**							R21
Semester	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
III	0	0	3	48	1.5	40	60	100

Pre-requisite: Knowledge of File and Record Structures, Data Structures

Course Objectives:

1. To populate and query a database using SQL DDL/DML Commands.
2. To design real-world entities with Entity-Relationship diagrams.
3. To apply integrity constraints over relational databases.
4. To construct queries using advanced concepts of SQL
5. To demonstrate programs in PL/SQL

Course Outcomes: After successful completion of the course, Student will be able to:

CO 1	Use SQL for creating database and performing data manipulation operations. (BL-3)
CO 2	Examine integrity constraints to build efficient databases. (BL-3)
CO 3	Sketch PL/SQL programs including procedures, functions, cursors and triggers.(BL-3)
CO 4	Apply queries using advanced database design and Normalization. (BL-3)

CO-PO Mapping

CO	PO												PSO	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO 1	PSO 2
CO1	3	3	3						2	2			2	2
CO2	3	3	3						2	2			2	2
CO3	3	2							2	2			2	2
CO4	3	3	3		2				2	2			2	2

1: Low, 2-Medium, 3- High

COURSE CONTENT	CO
Task - 1 BASIC CONCEPTS (3H)	
1.Create a table called Employee with the following structure. **Name** **Type** Empno Number Ename Varchar2(20) Job Varchar2(20) Mgr Number Sal Number	CO 1

a. Add a column commission with domain to the Employee table.
b. Insert any five records into the table.
c. Update the column details of job
d. Rename the column of Employ table using alter command.
e. Delete the employee whose empno is19.

2.Create department table with the following structure.

Name	**Type**
Deptno	Number
Deptname	Varchar2(20)
location	Varchar2(20)

a. Add column designation to the department table.
b. Insert values into the table.
c. List the records of emp table grouped by dept no.
d. Update the record where dept no is 9.
e. Delete any column data from the table

3. Create a table called Customer table

Name	**Type**
Custname	Varchar2(20)
Custstreet	Varchar2(20)
Cust city	Varchar2(20)

a. Insert records into the table.
b. Add salary column to the table.
c. Alter the table column domain.
d. Drop salary column of the customer table.
e. Delete the rows of customer table whose Cust_city is 'hyd'.
f. Create a table called branch table.

Name	**Type**
Branchname	Varchar2(20)
Branch city	Varchar2(20)
asserts	Number

4. Increase the size of data type for asserts to the branch.
a. Add and drop a column to the branch table.
b. Insert values to the table.
c. Update the branch name column
d. Delete any two columns from the table
5. Create a table called sailor table

Name **Type** Sid Number Sname Varchar2(20) rating Varchar2(20) a. Add column age to the sailor table. b. Insert values into the sailor table. c. Delete the row with rating>8. d. Update the column details of sailor. e. Insert null values into the table. 6. Create a table called reserves table **Name** **Type** Boatid Integer sid Integer day Integer a. Insert values into the reserves table. b. Add column time to the reserves table. c. Alter the column day data type to date. d. Drop the column time in the table. e. Delete the row of the table with some condition.	
Task 2 - QUERIES USING DDL AND DML	**(6H)**
1. a. Create a user and grant all permissions to the user. b. Insert the any three records in the employee table and use rollback. Check the result. c. Add primary key constraint and not null constraint to the employee table. d. Insert null values to the employee table and verify the result. 2. a. Create a user and grant all permissions to the user. b. Insert values in the department table and use commit. c. Add constraints like unique and not null to the department table. d. Insert repeated values and null values into the table. 3. a. Create a user and grant all permissions to the user. b. Insert values into the table and use commit. c. Delete any three records in the department table and use rollback. d. Add constraint primary key and foreign key to the table. 4. a. Create a user and grant all permissions to the user. b. Insert records in the sailor table and use commit. c. Add save point after insertion of records and verify save point. d. Add constraints not null and primary key to the sailor table.	CO 1

5. a. Create a user and grant all permissions to the user. b. Use revoke command to remove user permissions. c. Change password of the user created. d. Add constraint foreign key and notnull. 6. a. Create a user and grant all permissions to the user. b. Update the table reserves and use save point and rollback. c. Add constraint primary key, foreign key and not null to the reserves table d. Delete constraint not null to the table column	
Task -3 QUERIES USING AGGREGATE FUNCTIONS	**(3H)**
1. a. By using the group by clause, display the names who belongs to dept no 10 along with average salary. b. Display lowest paid employee details under each department. c. Display number of employees working in each department and their department number. d. Using built in functions, display number of employees working in each department and their department name from dept table. Insert dept name to dept table and insert dept name for each row, do the required thing specified above. e. List all employees which start with either B or C. f. Display only these ename of employees where the maximum salary is greaterthan or equal to 5000. 2. a. Calculate the average salary for each different job. b. Show the average salary of each job excluding manager. c. Show the average salary for all departments employing more than three people. d. Display employees who earn more than the lowest salary in department 30 e. Show that value returned by sign (n) function. f. How many days between day of birth to current date 3. a. Show that two substring as single string. b. List all employee names, salary and 15% rise in salary. c. Display lowest paid emp details under each manager d. Display the average monthly salary bill for each deptno. e. Show the average salary for all departments employing more than two people. f. By using the group by clause, display the eid who belongs to dept no 05 along with average salary. 4. a. Count the number of employees in department20 b. Find the minimum salary earned by clerk. c. Find minimum, maximum, average salary of all employees. d. List the minimum and maximum salaries for each job type.	CO2

e. List the employee names in descending order. f. List the employee id, names in ascending order by empid. 5. a. Find the sids, names of sailors who have reserved all boats called "INTERLAKE Find the age of youngest sailor who is eligible to vote for each rating level with at least two such sailors. b. Find the sname, bid and reservation date for each reservation. c. Find the ages of sailors whose name begin and end with B and has at least 3characters. d. List in alphabetic order all sailors who have reserved red boat. e. Find the age of youngest sailor for each rating level. 6. a. List the Vendors who have delivered products within 6 months from orderdate. b. Display the Vendor details who have supplied both Assembled and Subparts. c. Display the Sub parts by grouping the Vendor type (Local or Non Local). d. Display the Vendor details in ascending order. e. Display the Sub part which costs more than any of the Assembled parts. f. Display the second maximum cost Assembled part	
TASK-4 PROGRAMS ON PL/SQL	**(6H)**
1. a. Write a PL/SQL program to swap two numbers. b. Write a PL/SQL program to find the largest of three numbers. 2. a. Write a PL/SQL program to find the total and average of 6 subjects and display the grade. b. Write a PL/SQL program to find the sum of digits in a given umber. 3. a. Write a PL/SQL program to display the number in reverse order. b. Write a PL/SQL program to check whether the given number is prime or not. 4. a. Write a PL/SQL program to find the factorial of a given number. b. Write a PL/SQL code block to calculate the area of a circle for a value of radius varying from 3 to 7. Store the radius and the corresponding values of calculated area in an empty table named areas, consisting of two columns radius and area. 5. a. Write a PL/SQL program to accept a string and remove the vowels from the string.(When 'hello' passed to the program it should display 'Hll' removing e and o from the world Hello). b. Write a PL/SQL program to accept a number and a divisor. Make sure the divisor is less than or equal to 10. Else display an error message. Otherwise Display the remainder in nwords.	CO 3
TASK-5 PROCEDURES AND FUNCTIONS	**(3H)**
1. Write a function to accept employee number as parameter and return Basic +HRA together as single column. 2. Accept year as parameter and write a Function to return the total net salary	CO 3

spent for a given year.
3. Create a function to find the factorial of a given number and hence find NCR.
4. Write a PL/SQL block to print prime Fibonacci series using local functions.
5. Create a procedure to find the lucky number of a given birth date.
6. Create function to the reverse of given number

TASK-6 TRIGGERS **(3H)**

CO 3

1. Create a row level trigger for the customers table that would fire for INSERT or UPDATE or DELETE operations performed on the CUSTOMERS table. This trigger will display the salary difference between the old values and new values:

CUSTOMERS table:

ID	NAME	AGE	ADDRESS	SALARY
1	Alive	24	Khammam	2000
2	Bob	27	Kadapa	3000
3	Catri	25	Guntur	4000
4	Dena	28	Hyderabad	5000
5	Eeshwar	27	Kurnool	6000
6	Farooq	28	Nellore	7000

2. Creation of insert trigger, delete trigger, update trigger practice triggers using the passenger database.
Passenger (Passport_ id INTEGER PRIMARY KEY, Name VARCHAR (50) NotNULL, Age Integer Not NULL, Sex Char, Address VARCHAR (50) NotNULL);
a. Write a Insert Trigger to check the Passport_id is exactly six digits ornot.
b. Write a trigger on passenger to display messages ‘1 Record is inserted’, ‘1 record is deleted’, ‘1 record is updated’ when insertion, deletion and updation are done on passenger respectively.
3. Insert row in employee table using Triggers. Every trigger is created with name any trigger has same name must be replaced by new name. These triggers can be raised before insert, update or delete rows on data base. The main difference between a trigger and a stored procedure is that the former is attached to a table and is only fired when an INSERT, UPDATE or DELETE occurs.
4. Convert employee name into uppercase whenever an employee record is inserted or updated. Trigger to fire before the insert or update.
5. Trigger before deleting a record from emp table. Trigger will insert the row to be deleted into table called delete _emp and also record user who has deleted the record and date and time of delete.
6. Create a transparent audit system for a table CUST_MSTR. The system must keep track of the records that are being deleted or updated.

TASK-7 BOOK PUBLISHING COMPANY	(6H)
A publishing company produces scientific books on various subjects. The books are written by authors who specialize in one particular subject. The company employs editors who, not necessarily being specialists in a particular area, each take sole responsibility for editing one or more publications. A publication covers essentially one of the specialist subjects and is normally written by a single author. When writing a particular book, each author works with on editor, but may submit another work for publication to be supervised by other editors. To improve their competitiveness, the company tries to employ a variety of authors, more than one author being a specialist in a particular subject for the above case study, do the following: 1. Analyze the data required. 2. Normalize the attributes. 3. Create the logical data model using E-R diagrams	CO 3
TASK-8 GENERAL HOSPITAL	**(6H)**
A General Hospital consists of a number of specialized wards (such as Maternity, Pediatric, Oncology, etc.). Each ward hosts a number of patients, who were admitted on the recommendation of their own GP and confirmed by a consultant employed by the Hospital. On admission, the personal details of every patient are recorded. A separate register is to be held to store the information of the tests undertaken and the results of a prescribed treatment. A number of tests may be conducted for each patient. Each patient is assigned to one leading consultant but may be examined by another doctor, if required. Doctors are specialists in some branch of medicine and may be leading consultants for a number of patients, not necessarily from the same ward. For the above case study, do the following. 1. Analyze the data required. 2. Normalize the attributes. Create the logical data model using E-R diagrams	CO 3
TASK -9 CAR RENTAL COMPANY	**(6H)**
A database is to be designed for a car rental company. The information required includes a description of cars, subcontractors (i.e. garages), company expenditures, company revenues and customers. Cars are to be described by such data as: make, model, year of production, engine size, fuel type, number of passengers, registration number, purchase price, purchase date, rent price and insurance details. It is the company policy not to keep any car for a period exceeding one year. All major repairs and maintenance are done by subcontractors (i.e. franchised	CO 4

garages), with whom CRC has long-term agreements. Therefore, the data about garages to be kept in the database includes garage names, addresses, range of services and the like. Some garages require payments immediately after a repair has been made; with others CRC has made arrangements for credit facilities. Company expenditures are to be registered for all outgoings connected with purchases, repairs, maintenance, insurance etc. Similarly, the cash inflow coming from all sources: Car hire, car sales, insurance claims must be kept of file. CRC maintains a reasonably stable client base. For this privileged category of customers special credit card facilities are provided. These customers may also book in advance a particular car. These reservations can be made for any period of time up to one month. Casual customers must pay a deposit for an estimated time of rental, unless they wish to pay by credit card. All major credit cards are accepted. Personal details such as name, address, telephone number, driving license, number about each customer are kept in the database. For the above case study, do the following: 1. Analyze the data required. 2. Normalize the attributes. Create the logical data model using E-R diagrams	
TASK -10 STUDENT PROGRESS MONITORING SYSTEM	**(6H)**
A database is to be designed for a college to monitor students' progress throughout their course of study. The students are reading for a degree (such as BA, BA (Hons) M.Sc., etc)within the framework of the modular system. The college provides a number of modules, each being characterized by its code, title, credit value, module leader, teaching staff and the department they come from. A module is coordinated by a module leader who shares teaching duties with one or more lecturers. A lecturer may teach (and be a module leader for) more than one module. Students are free to choose any module they wish but the following rules must be observed: Some modules require pre- requisites modules and some degree programs have compulsory modules. The database is also to contain some information about students including their numbers, names, addresses, degrees they read for, and their past performance i.e. modules taken and examination results. For the above case study, do the following: 1. Analyze the data required. 2. Normalize the attributes. 3. Create the logical data model i.e., ER diagrams. 4. Comprehend the data given in the case study by creating respective tables with primary keys and foreign keys where ever required. 5. Insert values into the tables created (Be vigilant about Master- Slave tables).	CO 4

6. Display the Students who have taken M.Sc course 7. Display the Module code and Number of Modules taught by each Lecturer. 8. Retrieve the Lecturer names who are not Module Leaders. 9. Display the Department name which offers 'English' module. 10. Retrieve the Prerequisite Courses offered by every Department (with Department names). 11. Present the Lecturer ID and Name who teaches 'Mathematics'. 12. Discover the number of years a Module is taught. 13. List out all the Faculties who work for 'Statistics' Department. 14. List out the number of Modules taught by each Module Leader. 15. List out the number of Modules taught by a particular Lecturer. 16. Create a view which contains the fields of both Department and Module tables. (Hint- The fields like Module code, title, credit, Department code and its name). 17. Update the credits of all the prerequisite courses to 5. Delete the Module 'History' from the Module table.	

Additional Experiments:	
TASK -1 – PROCEDURES	
1. Create the procedure for palindrome of given number. 2. Create the procedure for GCD: Program should load two registers with two Numbers and then apply the logic for GCD of two numbers. GCD of two numbers is performed by dividing the greater number by the smaller number till the remainder is zero. If it is zero, the divisor is the GCD if not the remainder and the divisors of the previous division are the new set of two numbers. The process is repeated by dividing greater of the two numbers by the smaller number till the remainder is zero and GCD is found. 3. Write the PL/SQL programs to create the procedure for factorial of given number. 4. Write the PL/SQL programs to create the procedure to find sum of N natural number. 5. Write the PL/SQL programs to create the procedure to find Fibonacci series. 6. Write the PL/SQL programs to create the procedure to check the given number is perfect or not	CO 1
TASK -2 – CURSORS	
1. Write a PL/SQL block that will display the name, dept no, salary of fist highest paid employees**.** **2.** Update the balance stock in the item master table each time a transaction takes place in the item transaction table. The change in item master table depends on the item id is already present in the item master then update operation is performed to	CO 3

decrease the balance stock by the quantity specified in the item transaction in case the item id is not present in the item master table then the record is inserted in the item master table. 3. Write a PL/SQL block that will display the employee details along with salary using cursors. 4. To write a Cursor to display the list of employees who are working as a Managers or Analyst. 5. To write a Cursor to find employee with given job and dept no. 6. Write a PL/SQL block using implicit cursor that will display message, the salaries of all the employees in the 'employee' table are updated. If none of the employee's salary are updated, we geta message 'None of the salaries were updated'. Else we get a message like for example, 'Salaries for 1000 employees are updated' if there are 1000 rows in 'employee' table	

Virtual Labs:

http://vlabs.iitb.ac.in/vlabs-dev/labs/dblab/labs/explist.php

List of Experiments with Description:

1. Data Definition Language(DDL) Statements: (Create table, Alter table, Drop table)
   Aim: To Understand and Implement Data Defining Language (DDL) Statements.
   Objective: To understand the various aspects of Data definition language commands like:
   Creating a table, with or without constraints.
   Understanding Data types.
   Altering the structure of the table like adding attributes at later stage, modifying size of attributes or adding constraints to attributes.
   Removing the table created, i.e., Drop table in SQL.
2. Data Manipulation Language(DML) Statements
   Aim: To understand the concept of implementing Data Manipulation Language (DML) statements.
   The objective of the experiment is to understand various aspects of Data Manipulation Commands like:
   Inserting Data into the table, (inserting all attributes in a table or inserting selected attributes in a table).
   Updating Data into the table (updating all tuples in a table or updating selected tuples in a table).
   Deleting Data from the table (deleting all tuples from the table(not advisable) or deleting selected tuples from the table).
3. Data Query Language(DQL) Statements: (Select statement with operations like Where clause, Order by, Logical operators, Scalar functions and Aggregate functions)
   Aim: To understand various aspects of Data Query Language Commands like
   Displaying all the attributes and tuples from the table.
   Displaying selected attributes/tuples from the table.

Using Logical and comparison operators.
Using aggregate functions.
Using Scalar functions.
Sorting Data.

4. Transaction Control Language(TCL) statements: (Commit(make changes permanent), Rollback (undo)
   Aim: To understand and implement Transaction Control Language (TCL) Statements.
   Objective: To Provide the students a practical experience of how transactions could be made permanent in memory or how are they revoked.
5. Describe statement: To view the structure of the table created
   Aim: To understand and Implement Describe Statement which can be used to view the structure of the table created by the user.
   Procedure:
   The Describe command is used to view the structure of the table created.
   To use the describe statement, you should have at least one table in your schema.
   The syntax for describe is desc <table_name>
   Example : If you would like to view Employee table, then Desc emp;
   Write Query in the Query Editor and click on Execute Query button.
   If you are existing user and want to save/restore your data, use Credentials.

Text Book(s):

1. A. Silberschatz, H.F. Korth, S. Sudarshan, "Database System Concepts", 6/e, TMH 2019
2. Raghurama Krishnan, Johannes Gehrke, "Database Management Systems", 3/e, TMH

Reference Book(s):

1. RamezElmasri, Shamkant, B. Navathe, "Database Systems", Pearson Education, 6/e, 2013.
2. Peter Rob, Carles Coronel, "Database System Concepts", Cengage Learning, 7/e,2008.Rick F Vander Lans, "Introduction to SQL", 4/e, Pearson Education, 2007
3. Nilesh Shah, "Database Systems Using Oracle", PHI, 2007

NARAYANA ENGINEERING COLLEGE::GUDUR								
21CS2502	**OBJECT ORIENTED PROGRAMMING THROUGH JAVA LAB**							**R21**
Semester	**Hours / Week**			**Total hrs**	**Credit**	**Max Marks**		
	L	**T**	**P**		**C**	**CIE**	**SEE**	**TOTAL**
III	0	0	3	48	1.5	40	60	100

Pre-requisite: Programming knowledge

Course Objectives:

1. To understand fundamentals of programming such as variables, conditional and iterative execution, methods, etc.
2. To understand fundamentals of object-oriented programming in Java, defining classes, invoking methods, using class libraries, etc.
3. To develop programs on object-oriented programming concepts through java.
4. To develop programs on Exception Handling and multi-threading concepts.

Course Outcomes: After successful completion of the course, the student will be able to:

CO 1	Apply the fundamental elements of java programming to solve given problems.(BL-3)
CO 2	Implement the concepts of object oriented programming to solve the applications. (BL-3)
CO 3	Apply the Method overloading and exception handling mechanisms to solve given problems. (BL-3)
CO 4	Apply the Multithreading and packages to improve the system performance. (BL-3)

CO-PO Mapping

CO	**PO**												**PSO**	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	3	3	3										3	3
CO2	3	3	2		3								3	2
CO3	3	2	2		2								2	3
CO4	3	2	3		3								2	3

1: Low, 2-Medium, 3- High

COURSE CONTENT	CO
Task 1 - Basics	**(6H)**
a). Write a JAVA program to display default value of all primitive data type of JAVA? b). Write a java program that display the roots of a quadratic equation ax2+bx=0. Calculate the discriminate D and basing on value of D, describe the nature of root. ? c). Five Bikers Compete in a race such that they drive at a constant speed which may or may not be the same as the other. To qualify the race, the speed of a racer must be more than the average speed of all 5 racers. Take as input the speed of each racer and print back the speed of qualifying racers. ? d) Write a case study on public static void main (250 words) ?	CO 1
Task -2 Control-flow, Strings	**(4H)**
a). The Fibonacci sequence is defined by the following rule. The first two values in the sequence are 1 and 1. Every subsequent value is the sum of the two values preceding it. b) Write a java program to multiply two given matrices. c) Write a JAVA program using String Buffer to delete, remove character. ? d) Write a program to perform the following operations on strings through interactive input. 1) Sort given strings in alphabetical 2) Convert the strings to uppercase. ?	CO 1
Task -3 Class, Objects	**(4H)**
a). Write a JAVA program to implement class mechanism. – Create a class, methods and invoke them inside main method. ? b). Write a JAVA program to implement constructor. ?	CO 2
TASK-4 Methods	**(4H)**
a). Write a JAVA program to implement constructor overloading. ? b). Write a JAVA program implement method overloading. ?	CO 2
TASK-5 Inheritance	**(6H)**
a). Write a JAVA program to implement Single Inheritance? b). Write a JAVA program to implement multi level Inheritance? c). Write a java program for abstract class to find areas of different shapes?	CO 3
TASK-6 Interfaces	**(6H)**
a). Write a JAVA program give example for “super” keyword. ? b). Write a JAVA program to implement Interface. What kind of Inheritance can be achieved? c). Write a JAVA program to implement multiple inheritance access in java? d). Write a JAVA program by using extends and implements keywords?	CO 3
TASK-7 Exceptions	**(4H)**
a).Write a JAVA program that describes exception handling mechanism. ? b).Write a JAVA program Illustrating Multiple catch clauses?	CO 3
TASK-8 Runtime Polymorphism	**(4H)**
a). Write a JAVA program that implements Runtime polymorphism? b). Write a Case study on run time polymorphism, inheritance that implements in above problem?	CO 4
TASK-9 User defined Exception	**(6H)**

a). Write a JAVA program for creation of Illustrating throw? b). Write a JAVA program for creation of Illustrating finally? c). Write a JAVA program for creation of Java Built-in Exceptions? d).Write a JAVA program for creation of User Defined Exception?	CO 4
TASK -10 Threads	**(4H)**
a). Write a JAVA program that creates threads by extending Thread class .First thread display "Good Morning "every 1 sec, the second thread displays "Hello "every 2 seconds and the third display "Welcome" every 3 seconds ,(Repeat the same by implementing Runnable) ? b). Write a program illustrating isAlive and join ()? c). Create two threads such that one of the thread print even no's and another printsodd no's up to a given range. ?	CO 4
TASK-11 Threads continuity	**(4H)**
a).Write a JAVA program Producer Consumer Problem? b).Write a case study on thread Synchronization after solving the above producer consumer problem?	CO 4
TASK-12 Packages	**(4H)**
a). Write a JAVA program illustrate class path? b). Write a case study on including in class path in your os environment of your package.? c). Write a Java Program to Create a package called "Arithmetic" that contains methods to deal with all arithmetic operations. Also, write a program to use the package.?	CO 4

Additional Experiments:	
TASK-1 Applet	
a).Write a JAVA program to paint like paint brush in applet. ? b) Write a JAVA program to display analog clock using Applet. ? c). Write a JAVA program to create different shapes and fill colours using Applet. ? d). Write an applet illustrating sequence of events in an applet. ?	
TASK -2 Files	
a) Write a java program that reads a file name from the user, and 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. b) Write a java program that displays the number of characters, lines and words in a text file. c) Write a java program that reads a file and displays the file on the screen with line number before each line.	
Virtual Labs: 1. http://cse02-iiith.vlabs.ac.in/ 2. http://vlabs.iitb.ac.in/vlabs-dev/labs/java-iitd/experiments/java-intro-iitd/simulation.html	

Text Book(s):

1. Herbert Schildt "Java The complete reference", 9th edition, McGraw Hill Education (India) Pvt. Ltd.
2. Ivor Horton, Beginning Java 2, JDK 5th Edition, Wiley dreamtech.

Reference Book(s):

1. R AJohson-Thomson,An introduction to java programming and object oriented application development,
2. Y Daniel liang, Introduction to java programming 6th Edition, Pearson Education.
3. C.Xavier, Java programming: A practical approach, First edition,TMH, 2011.
4. Bruce Eckel, Thinking in Java, 2nd Edition, Pearson Education
5. H.M Dietel and P.J Dietel, Java How to Program, 6th Edition, Pearson Ed.
6. Y. Daniel Liang, Introduction to Java programming-comprehensive, Tenth Edition, Pearson ltd 2015.
7. E Balagurusamy, Programming With Java : A Primer 5th Edition Tata McGraw Hill.

SEMESTER-IV

NARAYANA ENGINEERING COLLEGE::GUDUR							
21MA1007	EXPLORATORY DATA ANALYTICS WITH R						R21
Semester	**Hours / Week**			**Total hrs**	**Credit**	**Max Marks**	
	L	T	P		C	CIE	SEE	TOTAL
IV	3	0	0	48	3	40	60	100

Pre-requisite: Engineering Mathematics, Computer Programming.

Course Objectives:

1. To understand the fundamentals of 'R' programming
2. To identify appropriate statistical tests.
3. To implement commonly used statistical methods
4. To perform graphical analysis in R
5. To explore data-sets for generating testable hypotheses

Course Outcomes: On successful completion of the course, the student will be able to:

CO 1	Demonstratethe fundamental knowledge of R-Programming concepts for solving the engineering applications (BL-2)
CO 2	Applydataobjects& probability commands for data manipulations (BL-3)
CO 3	Applydescriptivestatistics and data distribution commands for statistical analysis (BL-3)
CO 4	Analyzehypothesis testing & graphical analysis on different data-sets for testable hypothesis and virtualization (BL-4)
CO 5	Analyzecomplex analytical models using formula syntax and regression for data analysis (BL-4)

CO-PO Mapping

CO	PO												PSO	
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
CO1	3	3	2									2	3	2
CO2	3	3	3	1								2	3	2
CO3	3	3	3	2								2	3	2
CO4	3	3	3	3	2							2	2	2
CO5	3	3	3	3	2							2	2	2

1: Low, 2-Medium, 3- High

COURSE CONTENT

MODULE – 1	Introduction to R Programming	10H

Reading and Getting Data into R, Viewing Named Objects, Types of Data Items, The Structure of Data Items, Working with History Commands, Saving your Work in R. Control Statements, Arithmetic and Boolean Operators, Functions, Return Values, Environment and Scope Issues, Recursion.

At the end of the Module 1, students will be able to:

1. Understand the basics of R programming. (BL-2)
2. Demonstrate the working environment of R Programming. (BL-2)
3. Understand R programming elements. (BL-2)

MODULE – 2	Objects in R and Probability methods	10H

Manipulating Objects, Viewing Objects within Objects, Constructing Data Objects, Forms of Data Objects: Testing and Converting. Sample Spaces, Events, Properties of Probability, Counting Methods, Conditional Probability, Independent Events, Bayes' Rule, Random Variables.

At the end of the Module 2, students will be able to:

1. Construct data objects from the keyboard, clipboard, or external data files. (BL-3)
2. Demonstrate various commands for probability formulae. (BL-2)
3. Apply probability functions for problem solving in R. (BL-3)

MODULE – 3	Descriptive statistical analysis	10H

Summary Commands, Summarizing Samples, Summary Tables.Creating Data for Complex Analysis, Summarizing Data.Stem and Leaf Plot, Histograms, Density Function, Types of Data Distribution, The Shapiro-Wilk Test for Normality, The Kolmogorov-Smirnov Test, Quantile-Quantile Plots

At the end of the Module 3, students will be able to:

1. Demonstrate summary commands on data, Stem and Leaf Plot & Histograms. (BL-2)
2. Apply commands for complex data analysis and summarize the data. (BL-3)
3. Demonstrate the test methods in R programming. (BL-2)

MODULE – 4	Hypothesis Testing & Graphical Analysis	9H

Using the Student's t-test, The Wilcoxon U-Test (Mann-Whitney), Paired t- and U-Tests, Correlation and Covariance, Tests for Association. Box-whisker Plots, Scatter Plots, Pairs Plots (Multiple Correlation Plots) Line Charts, Pie Charts, Cleveland Dot Charts, Bar Charts, Copy Graphics to Other Applications.

At the end of the Module 4, students will be able to:

1. Apply the methods of testing hypotheses using standard tests. (BL-3)
2. Create summary tables, cross-tabulate. (BL-2)
3. Conduct test for non-parametric data, paired tests for parametric and non-parametric data. (BL-2)
4. Describe generating correlation and covariance matrices. (BL-2)

MODULE – 5	Complex Statistical analysis and Regression	9H

Examples of Using Formula Syntax for Basic tests, Formula Notation in Graphics, Analysis of Variance (ANOVA).Simple Linear Regression, Multiple Regression, Curvilinear Regression, Plotting Linear Models and Curve Fitting, Summarizing Regression Models.

At the end of the Module 5, students will be able to:

1. Create a range of graphs to summarize your data and results. (BL-2)
2. Illustrate box-whisker plots, scatter plots, including multiple correlation plots. (BL-3)
3. Move graphs from R to other programs and save graphs as files on disk. (BL-2)
4. Explain formula notation for simple hypothesis tests, graphics. (BL-2)

Total hours:	**48 hours**

Content beyond syllabus: Linear Algebra Operations on Vectors and Matrices, Set Operations, Writing own scripts, Building R Packages

Self-Study:

Contents to promote self-Learning:

SN.O	Module	Reference

1	Calculations with R Software	https://nptel.ac.in/courses/111/104/111104120/ **Lecture -1, 2, 3, 4, 5**
2	Introduction to Descriptive statistics, frequency distribution	https://nptel.ac.in/courses/111/104/111104120/ **Lecture -6, 7, 8, 9**
3	Graphics and plots	https://nptel.ac.in/courses/111/104/111104120/ **Lecture -10, 11, 12, 13**
4	Central tendency of data	https://nptel.ac.in/courses/111/104/111104120/ **Lecture -14, 15, 16, 17, 18**
5	Variation in data Association of variables, Regression	https://nptel.ac.in/courses/111/104/111104120/ **Lecture -19, 20, 21** https://nptel.ac.in/courses/111/104/111104120/ **Lecture -22-28** https://nptel.ac.in/courses/110/107/110107092/ **Lecture -21-25**

Text Book(s):

1. Mark Gardener, Beginning R The Statistical Programming language- John Wiley & Sons, Inc, 2016
2. G J KERNS, Introduction to Probability and Statistics Using R, 1st edition, GNU Free Documentation License, 2010

Reference Book(s):

1. Norman Matloff, The Art of R Programming, A Tour of statistical software design, NSP, 2011
2. Michael J. Crawley, The R Book, WILEY, 2012.
3. John Maindonald, W. John Braun, Data Analysis and Graphics Using R, Third Edition, Cambridge University Press, 2010
4. Roger D. Peng and Elizabeth Matsui, The Art of Data Science- A Guide for anyone Who Works with Data –Leanpub Publications, 2014
5. Grolemund, Garrett, Hands-On Programming with R Paperback by SPD,2014
6. Prabhanjan Narayanachar Tattar, Suresh Ramaiah, B.G. Manjunath, A Course in statistics with R, 1st edition, Wiley, 2016
7. Braun W. J., Murdoch D. J., A First Course in Statistical Programming with R, Cambridge University Press, 2007

Online Resources/ Web References:

1. https://www.edx.org/course/data-science-r-basics
2. https://www.coursera.org/specializations/statistics
3. https://www.coursera.org/learn/r-programming
4. https://www.youtube.com/watch?v=7076ZuAwUn8&list=PLWPirh4EWFpEvN4ktS8LE0cvLCSfhD55t
5. https://rextester.com/l/r_online_compiler
6. https://www.r-project.org/about.html
7. https://www.datamentor.io/r-programming/
8. https://www.tutorialspoint.com/r/index.htm
9. https://www.w3schools.in/r/

NARAYANA ENGINEERING COLLEGE::GUDUR								
21CS2005	**COMPUTER NETWORKS**							R21
Semester	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
IV	3	0	0	48	3	40	60	100

Pre-requisite: Knowledge of Information Technology, Computer Organization & Architecture

Course Objectives:

1. To impart the core principles of Information Communication Technology.
2. To deliver background information on the key transmission technologies used in computer networks.
3. To convey dimensions of Network layer through Internet Protocol.
4. To provide an insight into the most widely used Transport Layer protocols
5. To teach the principles of Application Layer and its protocols.

Course Outcomes: On successful completion of the course, student will be able to:

CO 1	Describe the concepts of Internet in terms of its building blocks, organized layered architecture. (BL-2)
CO 2	Identify the errors in data transfer between source and destination. (BL-2)
CO 3	Demonstrate the skills of sub netting and routing protocols. (BL-3)
CO 4	Illustrate the reliable, unreliable communication on public networks for various applications. (BL-3)
CO 5	Explain the principles of Application Layer and its protocols(BL-4).

CO-PO Mapping

CO	PO												PSO	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	3	2	2										3	2
CO2	3	3	3										3	2
CO3	3	3	3										3	2
CO4	3	3	3										3	3
CO5	3	3	2										3	3

1: Low, 2-Medium, 3- High

COURSE CONTENT		
MODULE – 1	**Physical Layer**	**10H**
Data Communications, Networks, Network Types, Internet History, Standards and Administration, Protocol Layering, TCP/IP Protocol Suite, The OSI Model**.** Data and Signals, Digital Signals, Transmission Impairment, Data Rate Limits, Performance. Transmission Media: Introduction, Guided Media, Unguided Media		
MODULE – 2	**Data-Link Layer & MAC**	**9H**
Introduction, Link-Layer Addressing, Error Detection and Correction: Cyclic Codes, Checksum, Forward Error Correction, Data Link Control (DLC)**:**DLC Services, Data-Link Layer Protocols, Sliding Window Protocols, HDLC, PPP.MAC: Random Access.		
MODULE – 3	**Network Layer**	**10H**
Network Layer: Network Layer Design Issues, Routing Algorithms: The Optimality Principle, Shortest Path Algorithm, Flooding, Distance Vector, Link State, Hierarchical, Broadcast, Multicast, Anycast, Congestion Control Algorithms, Quality of Service. Internetworking, IPV4 Addresses, IPV6, OSPF, BGP, IP.		
MODULE – 4	**Transport Layer**	**(9H)**
The Transport layer services, Elements of Transport Protocols, Congestion Control in Transport Layer. UDP, TCP, Performance problems in computer networks, Network performance measurement, Real-time interactive protocols.		
MODULE – 5	**Application Layer**	**10H**
Introduction, Client Server Programming-Iterative communication using UDP, Iterative communication using TCP. Standard Client Server Protocols: WWW, HTTP, Domain Name System, FTP, e-mail, TELNET, Secure Shell.		
	Total hours:	**48 hours**

Text Book(s):

1. Behrouz A. Forouzan, Data communications and networking, 5th edition, Mc Graw Hill Education, 2012.
2. Andrew S. Tanenbaum, Wetherall, Computer Networks, 5th edition, Pearson, 2013.

Reference Book(s):

1. Douglas E. Comer, Internetworking with TCP/IP – Principles, protocolsand architecture-Volume 15th edition, PHI.
2. Kurose James, Ross Keith, Computer Networking: A Top-Down Approach, 6th Edition, Pearson Education
3. Fall, Richard, TCP/IP Illustrated: The Protocols, 2ND edition, Pearson Education
4. Behrouz A. Forouzan, TCP/IP Protocol Suite, 4th edition, Tata McGraw Hill
5. Bhushan Trivedi, Data Communication and Networks, Oxford, 2016.
6. Davie, Elsevier,Computer Networks, 5th Edition, Peterson.
7. M. Dave, Computer Networks, Cengage Learning, 2012.

NARAYANA ENGINEERING COLLEGE::GUDUR								
21CS2006	**OPERATING SYSTEMS**							R21
Semester	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
IV	3	0	0	48	3	40	60	100

Pre-requisite: Fundamentals of computers

Course Objectives:

1. To understand the fundamental principles of the operating system, its services and Functionalities.
2. To illustrate the concepts of inter-process communication, synchronization and scheduling.
3. To understand different types of memory management viz. virtual memory, paging and segmentation.
4. To identify the reasons for deadlock and understand the techniques for deadlock detection, prevention and recovery.
5. To understand the need of Mass storage and protection mechanisms in computer systems.

Course Outcomes: After successful completion of the course, Student will be able to:

CO 1	Illustrate the concepts and design of operating system of a computer. (BL-2)
CO 2	Analyze CPU process scheduling and deadlock handling techniques provided with concurrencies. (BL-4)
CO 3	Analyze the memory management and virtual memory concepts of an application. (BL-4)
CO 4	Demonstrate the structure and implementation of file system for effective storage in a system. (BL-2)
CO 5	Illustrate Mass Storage Structure and Protection Mechanism of a system. (BL-2)

CO-PO Mapping														
CO	PO												PSO	
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
CO1	3	3	2										3	2
CO2	3	3	3										3	3
CO3	3	3	3										3	3
CO4	3	3	3										3	3
CO5	3	3	3										3	2
1: Low, 2-Medium, 3- High														

COURSE CONTENT		
MODULE – 1	**INTRODUCTION**	**9H**
Computer system architecture, operating systems structure, operating systems operations; Evolution of operating systems: Simple Batch, multi programmed, time shared, parallel distributed systems, real time systems, special purpose systems, operating system services, user operating systems interface. Types of systems calls, system programs, protection and security, operating system design and implementation, operating systems structure**.**		
MODULE – 2	**PROCESS AND CPU SCHEDULING, PROCESS COORDINATION**	**10H**
The process, process state, process control block, threads; Scheduling queues, context switch, pre-emptive scheduling, dispatcher, scheduling criteria, scheduling algorithms. Process synchronization, the critical section problem, synchronization hardware, semaphores and classic problems of synchronization monitor. Deadlock characterization, methods of handling deadlocks, deadlock prevention, dead lock avoidance, dead lock detection and recovery from deadlock**.**		
MODULE – 3	**MEMORY MANAGEMENT AND VIRTUAL MEMORY**	**10H**
Swapping, contiguous memory allocation, paging, structure of page table. Segmentation with paging, virtual memory, demand paging; Performance of demand paging: Page replacement, page replacement algorithms, allocation of frames, thrashing**.**		

MODULE – 4	FILE SYSTEM INTERFACE	(9H)
The concept of a file, access methods, directory structure, file system mounting, file sharing, protection, file system structure. File system structure, File system implementation, directory implementation, allocation methods, free space management**.**		
MODULE – 5	**MASS-STORAGE STRUCTURE**	**10H**
Overview of mass storage structure, Disk structure, Disk attachment, Disk scheduling, Disk management, Swap space management, RAID structure, Stable storage implementation**.** goals of protection, principles of protection, domain of protection, access matrix, implementation of access matrix		
	Total hours:	**48 hours**

TEXTBOOK:

1. Abraham Silberschatz, Peter B. Galvin, Greg Gagne, "Operating System Principles",10th Edition, Wiley Student Edition, 2018.
2. William Stallings, "Operating System- Internals and Design Principles", 6th Edition, Pearson Education, 2002.

REFERENCES:

1. D. M. Dhamdhere, "Operating Systems a Concept based Approach", 2nd Edition, Tata McGraw-Hill, 2006.
2. P.C.P. Bhatt, "An Introduction to Operating Systems", PHI Publishers.
3. G. Nutt, N. Chaki and S. Neogy, "Operating Systems", Third Edition, Pearson Education. Andrew S Tanenbaum, "Modern Operating Systems", 3rd Edition, PHI, 2007.

NARAYANA ENGINEERING COLLEGE:GUDUR								
21CS2007	**SOFTWARE ENGINEERING**							R21
Semester	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
IV	3	0	0	48	3	40	60	100

Pre-requisite: Programming Skills

Course Objectives:

1. To understand the software life cycle models.
2. To understand the software requirements and SRS document.
3. To understand the important of modeling and modeling languages
4. To design and develop correct and robust software products
5. To understand the maintenance of the software.

Course Outcomes: After successful completion of the course, Student will be able to:

CO 1	Understand Fundamental concepts of software engineering and analyze process models required to develop a software system.(BL-2)
CO 2	Analyze software requirements and model requirements for developing the application.(BL-4)
CO 3	Apply software design and development technique uses by understanding software architecture.(BL-3)
CO 4	Analyze the User interface design techniques to design GUI.(BL-4)
CO 5	Analyze the testing strategies and techniques for quality software.(BL-4)

CO-PO Mapping

CO	**PO**												**PSO**	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	3												2	2
CO2	2	3	3	1									3	2
CO3	3	3											2	2
CO4	3	3	2										3	3
CO5	3	3									3		3	2

1-Low, 2- Medium, 3- High

COURSE CONTENT		
MODULE – 1	**THE SOFTWARE PROCESS**	**10H**
The Nature of Software, The Unique Nature of Web Apps, Software Engineering, The Software Process, Software Engineering Practice, Software Myths**.** A Generic Process Model, Process Assessment and Improvement, Prescriptive Process Models, Specialized Process Models, The Unified Process, Personal and Team Process Models, Process Technology, Product and Process. Agility and the Cost of Change, Agile Process, Extreme Programming, Other Agile Process Models**.**		
MODULE – 2	**MODELING CONCEPTS**	**10H**
Class Diagrams, Deployment Diagrams, Use-Case Diagrams, Sequence Diagrams, Communication Diagrams, Activity Diagrams, State Diagrams. Requirements Engineering, Eliciting Requirements, Developing Use Cases, and Building the requirements model, Negotiating Requirements, Validating Requirements. Requirements Analysis, Scenario-Based Modeling, UML Models that Supplement the Use Case, Data Modeling Concepts, Class-Based Modeling**.**		
MODULE – 3	**DESIGN CONCEPTS**	**10H**
Design with Context of Software Engineering, The Design Process, Design Concepts, The Design Model. Software Architecture, Architecture Genres, Architecture Styles, Architectural Design, Assessing Alternative Architectural Designs, Architectural Mapping Using Data Flow. Component, Designing Class-Based Components, Conducting Component-level Design, Designing Traditional Components, Component-Based Development**.**		
MODULE – 4	**USER INTERFACE DESIGN, CODING AND TESTING**	**(9H)**
Characteristics of a Good User Interface, Basic Concepts, Types of User Interfaces, Fundamentals of Component-based GUI Development, A User Interface Design Methodology. Coding, Code Review, Software Documentation, Testing, Unit Testing, Black-box Testing, White-Box Testing		
MODULE – 5	**SOFTWARE QUALITY & PRODUCT METRICS**	**9H**
Software Quality, Software Quality Management System, ISO 9000, SEI Capability Maturity **Model Product metrics:** Metrics for Requirements Model, Metrics for Design Model, Metrics for source code, Metrics for testing, Metrics for maintenance.		
	Total hours:	**48 hours**

TEXTBOOK:

1. Roger S. Pressman, Software engineering A practitioner's Approach, Seventh Edition, McGraw Hill International Education, 2016.
2. Rajib Mall, Fundamentals of Software Engineering, Third Edition, PHI.

REFERENCES:

1. Ian Somerville, Software Engineering, 9th Edition Pearson Education Asia,2011.
2. Pankaj Jalote, A concise introduction to software Engineering, Springer
3. Pankaj Jalote, Software Engineering, A Precise Approach, Wiley India,2010
4. Jim Arlow, Ila Neustadt, UML 2 and the Unified Process: Practical Object-Oriented Analysis and Design, 2nd Edition, Pearson, (2005).

1. K.K. Agarwal & Yogesh Singh, Software Engineering, New Age International Publishers, 2007

NARAYANA ENGINEERING COLLEGE::GUDUR								
21MA1501	**EXPLORATORY DATA ANALYTICS WITH R LABORATORY**							R21
Semester	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
IV	0	0	3	48	1.5	40	60	100

Pre-requisite: Knowledge of Computer Programming, Probability and Statistics

Course Objectives:

6. To setup R tools and get familiarize with commands
7. To Execute commands related to Probability
8. To implement statistical analysis functions.
9. To draw graphs for the results in R Programming

Course Outcomes: After successful completion of the course, Student will be able to:

CO 1	Apply R elements for solving basic engineering applications (BL-3)
CO 2	Apply dataobjects commands for data manipulations (BL-3)
CO 3	Implement hypothesis testing &graphical Analysison different data-sets for testable hypothesis and virtualization (BL-3)
CO 4	Design and Implement engineering application usingANOVA for data analysis(BL-3)

CO-PO Mapping

CO	PO												PSO	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	3	3	3									2	3	2
CO2	3	3	3									2	3	2
CO3	3	3	3	2	1							2	3	2
CO4	3	3	3	2	1							2	3	2

1: Low, 2-Medium, 3- High

COURSE CONTENT	CO
TASK -1 Installing Packages (3H)	
Installing R tools and Exploring packages in R. Managing user workspace	CO 1
TASK -2 Basic Programs (3H)	
Programs on data types in R. Programs on Creating and manipulating a vector in R.	CO 1
TASK -3 Operations (3H)	
Programs on Creating matrix operations in R Programs on manipulating matrix in R. Programs on Creating and operations on Factors in R.	CO 1
TASK -4 Data Frames and Operators (6H)	
Programs on Data Frames in R. Programs on Operators in R. Programs on Data Sets.	CO 2
TASK -5 Working with Graphs (6H)	
Programs on Customizing and Saving to Graphs in R. Programs on PLOT Function in R to customize graphs Programs for Generating Boxplots, and Scatterplots	CO 2
Task - 6 Data distribution (6H)	
Programs on Random Number Generation and Control Programs on Random Numbers and Sampling Programs on Creating Random Data Partitions	CO 3
Task -7 Hypothesis Testing(3H)	
Programs on Simple Hypothesis Testing Programs on Correlation and Covariance.	CO 3
Task -8 ANOVA (6H)	
Simple Programs on Analysis of Variance (ANOVA) Programs on One-Way ANOVA Programs on Two-Way ANOVA	CO3
Task -9 ANOVA (6H)	
Programs for Performing simple Linear Regression. A. Give Me a Number - Regression B. Computing the Root-Mean-Square Error C. Performing Variable Selection in Linear Regression.	CO 3
Task -10 Data Summary (6H)	
Programs on Extracting Means Programs on Creating Standard Data Summaries Programs on Summary Statistics	CO 4

Additional Experiments: **TASK-1 Complex Analysis**	
Programs on Manipulating Data and Extracting Components Programs on Creating Data for Complex Analysis, Summarizing Data.	CO 4
TASK -2 Multiple Regression	

Programs on Multiple Regression Building Regression Trees	CO 4

Virtual Labs

1. https://app.cybrary.it/browse/next-tech-course/transfer-learning-r-programming?queryID=4c4829fb170457c5c2c5cff546ef2cf5&objectID=46375

In this virtual lab, you will learn the fundamentals of the R programming language, one of the most common programming languages utilized by data scientists and machine learning engineers. In this introductory lab you will learn the basics of objects, strings, data, and expressions for use in R.

List of Experiments:

1.1 Quick Start
1.2 Basic Objects
1.3 Managing Your Workspace
1.4 Basic Expressions
1.5 Working with Basic Objects
1.6 Working with Strings
1.7 Working with Data

2.https://app.cybrary.it/browse/next-tech-course/transfer-exploratory-data-analysis-in-r?queryID=7a61f9add7d43824dbbb5ca78171278c&objectID=46289

In this virtual lab, we will take a deeper dive into R in order to conduct some exploratory data analysis to convert structured data into usable models/charts for analysis. This will cover critical topics in R and data science such as data set extraction, data partitions, and data visualization

List of Experiments:

2.1 What's in There - Exploratory Data Analysis
2.2 Creating Standard Data Summaries
2.3 Extracting a Subset of a Dataset
2.4 Splitting a Dataset
2.5 Creating Random Data Partitions
2.6 Generating Standard Plots, such as Histograms, Boxplots, and Scatterplots
2.7 Generating Multiple Plots on a Grid
2.8 Creating Plots with the `lattice` Package
2.9 Creating Charts that Facilitate Comparisons
2.10 Creating Charts That Help to Visualize Possible Causality

3.https://app.cybrary.it/browse/next-tech-course/transfer-regression-analysis-in-r?queryID=655394865504019e0f9b3fb59c3cb66e&objectID=46430

In this virtual lab, you will utilize foundational knowledge of R in order to approach machine learning model driven regression analysis solutions to validate and measure the performance of said models. More specifically, we will cover linear regression, neural networks, regression trees, variable selection, and more.

List of Experiments:

3.1 Give Me a Number - Regression

3.2 Computing the Root-Mean-Square Error
3.3 Building KNN Models for Regression
3.4 Performing Linear Regression
3.5 Performing Variable Selection in Linear Regression
3.6 Building Regression Trees
3.7 Building Random Forest Models for Regression
3.8 Using Neural Networks for Regression
3.9 Performing K-Fold Cross-Validation
3.10 Performing Leave-One-Out Cross-Validation to Limit Overfitting

Text Book(s):

1. Beginning R The Statistical Programming language- Mark Gardener, John Wiley & Sons,Inc, 2015
2. The Art of R Programming, A Tour of statistical software design, Norman Matloff, NSP, 2011
3. Introduction to Probability and Statistics Using R, G J KERNS, 1st edition, GNU Free Documentation License, 2010

Reference Book(s):

1. Data Analysis and Graphics Using R, Third Edition, John Maindonald, W. John Braun, Cambridge University Press, 2010
2. Exploratory Data Analysis with R – Roger D. Peng, Leanpub publications, 2015
3. Introduction to Probability and Statistics Using R, G. jay Kerns, First Edition, 2011
4. The Art of Data Science- A Guide for anyone Who Works with Data – Roger D. Peng and Elizabeth Matsui, Leanpub Publications, 2014
5. Hands-On Programming with R Paperback by Grolemund (Author), Garrett (Author), SPD,2014
6. A Course in statistics with R, PrabhanjanNarayanacharTattar, Suresh Ramaiah, B.G. Manjunath, 1st edition, Wiley, 2016
7. A First Course in Statistical Programming with R, Braun W. J., Murdoch D. J., Cambridge University Press, 2007

Web References:

1. https://rextester.com/l/r_online_compiler (online compiler)
2. https://www.r-project.org/about.html
3. https://www.datamentor.io/r-programming/
4. https://www.tutorialspoint.com/r/index.htm
5. https://www.w3schools.in/r/
6. https://app.cybrary.it

NARAYANA ENGINEERING COLLEGE:: GUDUR								
21CS2503	**OPERATING SYSTEMS AND COMPUTER NETWORKS LAB**							R21
Semester	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
IV	0	0	3	48	1.5	40	60	100

Pre-requisite: Knowledge of Computer Programming, Information Technology.

Course Objectives:

10. To demonstrate the working principle of various communication protocols.
11. To implement data link layer and Network layer protocols.
12. To implement various CPU Scheduling, Deadlock Avoidance and detection Algorithms
13. To implement Page Replacement, File Organization and File Allocation Algorithms.

Course Outcomes: After successful completion of the course, the student will be able to:

CO 1	Analyze and simulate CPU Scheduling Algorithms like FCFS, Round Robin, SJF, Priority and Dead lock detection, avoidance (BL-3)
CO 2	Implement memory management schemes, page replacement schemes and File Organization techniques (BL-3)
CO 3	Analyze the concept of data link layer to differentiate Error detection and Correction codes for a computer network. (BL - 4)
CO 4	Analyze the concept of Network layer to differentiate various routing protocols for a network. (BL - 4)

CO-PO Mapping

CO	PO												PSO	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	2	3	3										3	2
CO2		3	3										3	2
CO3	3	3	3										3	2
CO4	3	3	3										3	3

1: Low, 2-Medium, 3- High

Operating Systems	
Task -1 (3H)	
Write a C program to simulate the following non-preemptive CPU Scheduling algorithms to find turnaround time and waiting time. (a) FCFS (b) SJF	CO 1
Task -2 (3H)	
Write a C program to simulate the following non-preemptive CPU Scheduling algorithms to find turnaround time and waiting time. (a) Round Robin (b) Priority	CO 1
Task -3 (3H)	
Write a C program to simulate Bankers algorithm for the purpose of deadlock avoidance	CO 1
TASK-4 (3H)	
Write a C program to simulate Bankers algorithm for the purpose of deadlock Prevention	CO 1
TASK-5 (3H)	
Write a C program to simulate page replacement algorithms FIFO	CO 2
TASK-6 (3H)	
Write a C program to simulate page replacement algorithms LRU	CO 2
TASK-7 (3H)	
Write a C program to simulate page replacement algorithms LFU	CO 2
TASK-8 (3H)	
Write a C program to simulate the MVT and MFT memory management techniques.	CO 2
TASK -9 (3H)	
Simulate paging technique of memory management	CO 2

Additional Experiments: (Operating Systems)	
TASK -1	
Write a C program to simulate the following file allocation strategies.	CO 2

(a) Sequential (b) Indexed (c) Linked	
TASK -2	
Write a C program to simulate the following file organization techniques (a) Single level directory (b)Two level directory	CO 2
TASK -3	
Write a C program to simulate the following file organization techniques (a) Hierarchical (b) DAG	CO 2

Virtual Labs:

http://vlabs.iitkgp.ernet.in/ant/

The Advanced Network Technologies Virtual Lab has been developed by keeping in mind the following objectives:

- To impart state-of-the-art knowledge on advanced topics in Computer Networks in an interactive manner through the Web
- Introduce the concept of network simulation to the students
- Involve students in analytical studies of Computer Networks through network simulation

All the while it is intended to present Computer Networks as an interesting subject to the students where learning and fun can go alongside.

http://vlabs.iitb.ac.in/vlabs-dev/vlab_bootcamp/bootcamp/CRUX/labs/index.html

1. Round Robin Process Scheduling Algorithm

http://vlabs.iitb.ac.in/vlabs-dev/vlab_bootcamp/bootcamp/CRUX/labs/exp1/index.html

COURSE CONTENT	CO
Computer Networks	
Task 1 - Framing methods (3H)	
Implement the following data link layer framing methods (a) Bit stuffing. (b) Character stuffing	CO 3
Task - 2 Encoding & Decoding (3H)	
Write a program to compute CRC code for the polynomials CRC-12, CRC-16	CO 3
Task -3 Sliding window protocols (3H)	
Develop a simple data link layer protocol that performs the flow control using the sliding window protocol, and loss recovery using the Go-Back-N mechanism	CO 3
TASK -4 Dijsktra's algorithm (3H)	
Implement Dijsktra's algorithm to compute the shortest path through a network	CO 4
TASK -5 Distance vector routing (3H)	
Implement distance vector routing algorithm for obtaining routing tables at each node	CO 4
TASK-6 Open Shortest Path First (3H)	
Implement distance vector routing algorithm for obtaining routing tables at each node	CO 4
TASK -7 Leaky bucket algorithm (3H)	
Write a program for congestion control using Leaky bucket algorithm.	CO 4
Additional Experiments:	
TASK -1 TCP Client server Programming	
Implement TCP Client server communication	CO 3
TASK -2 UDP Client server Programming	CO 3
Implement UDP Client server communication	

Text Book(s):

3. Behrouz A. Forouzan, Data communications and networking, Mc Graw Hill Education, 5th edition, 2012.
4. Andrew S. Tanenbaum, Wetherall, Computer Networks, Pearson, 5th edition, 2010.

Reference Book(s):

1. Douglas E. Comer, Internetworking with TCP/IP – Principles, protocols, and architecture-Volume 1, 5th edition, PHI
2. P.C.P Bhatt, An Introduction to Operating Systems, 2nd edition, PHI.
3. Douglas E. Comer, TCP/IP Client-Server Programming and Applications-Volume III, 2nd edition, Pearson
4. Kevin r fall, Richard, TCP/IP Illustrated: The Protocols, Volume 1, 2e, 2014, Pearson
5. Andrew S Tanenbaum, Modern Operating Systems 3rd Edition, PHI

NARAYANA ENGINEERING COLLEGE::GUDUR								
21CS2504	SOFTWARE ENGINEERING LAB							R21
Semester	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
IV	0	0	3	48	1.5	40	60	100

Pre-requisite: Problem solving skills

Course Objectives:

1. To gain knowledge on various tools for applying it in the software modeling and implementation.
2. To prepare students for performing requirement analysis and design of variety of applications.
3. To prepare students for project management.

Course Outcomes: After successful completion of the course, Student will be able to:

CO 1	Select suitable software development process model for the given scenario(BL-3)
CO 2	Classify the requirements and prepare software requirements specification for projects and perform modeling (BL-2)
CO 3	Understand the various design techniques and implement (BL-2)
CO 4	Apply testing principles for validating software project.(BL-3)

CO-PO Mapping

CO	PO												PSO	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO1**	**PSO 2**
CO1	2	2	1	1									2	2
CO2			2	2									2	2
CO3	1	1	1	1							1		2	2
CO4	1	1	1	1									2	2

1: Low, 2-Medium, 3- High

COURSE CONTENT		
TASK-1	**ROLE OF SOFTWARE**	**6H**
Objective: To identify the role of software in today's world across various domains. Software is also a predominant are for trade and export especially for the countries like India. Domains like health care, Airlines , financial Services, Insurance, retails, Education, and many more have exploited software and still there a lot of the scope for software to create impact and add values in multiple dimensions. Problem Description: In the context of this background, identify the areas (or application or systems) how software has been leveraged extensively in the following domains 1. Health Care 2. Airlines 3. Banking Insurance 4. Retail 5. Education Summary Identify the role of software across multiple domains related to day to day life.		
TASK– 2	**SOFTWARE DEVELOPMENT LIFE CYCLE MODELS**	**6H**
Objective: To identify the suitable process model. Justify the best suitable SDLC for the following: a. College automation system b. online shopping		
TASK-3	**SOFTWARE REQUIREMENTS SPECIFICATION**	**6H**
Draw use case diagram for Online Movie ticket reservation. Prepare use case diagram for Online airline reservation system		
TASK-4	**DATA MODELLING**	**6H**
Draw use case diagram for Online Movie ticket reservation. Prepare use case diagram for Online airline reservation system		
TASK-5	**CLASS MODELLING**	**6H**
Draw class diagram for Health care center. Draw class diagram for inventory system.		
TASK-6	**DATA MODELLING**	**6H**
Draw the class and use case diagram for Hospital management system?		
TASK-7	**SOFTWARE TESTING**	**3H**
Write the test cases for Banking application		
TASK-8	**SOFTWARE TESTING**	**3H**

Create a test plan documentation for Library management system.		
TASK-9	**SOFTWARE TESTING**	**3H**
UML Diagrams for develop the AUTOMATED TELLER MACHINE (ATM) application		
TASK-10	**SOFTWARE TESTING**	**3H**
UML Diagrams for develop the LIBRARY INFORMATION SYSTEM application.		
Additional Experiments:		
TASK-1	**SOFTWARE METRICS**	
Take ATM system study its system specification and report various bugs		
TASK -2	**SOFTWARE DESIGN**	
A program written in c language for Matrix multiplication fails. Introspect the causes for failure and write down the possible reasons for failure		
	Total Hours	**45 hours**

Virtual Labs:

http://vlabs.iitkgp.ernet.in/se/

To draw activity flow diagram for Library information system.
Draw a sequence diagram for Library information system.
Draw a state chart diagram for Library information system.
Write the test suites for user login functionality for library management system.
Determine the Cyclomatic complexity for the "ReissueBook" method as shown below:

```
public ID ReissueBook(ID userID, ID bookID) {

Member user = Member.GetMember(userID);

ID transactionID = null;

if ( user.canIssueNow() &&Book.IsAvailable(bookID) ) {

Integer count = user.getReissueCountFor(bookID);    // # of times this books has been reissued after it's recent issue by the user

if ( count < REISSUE_LIMIT ) {

user.incrementReissueCount(bookID);
BookTransaction transaction = new BookTrasaction(userID, bookID);
```

```
transaction.save();
transactionID = transaction.getID();
}
}
return transactionID;
```

Text Book(s):

1. Roger S. Pressman, "Software engineering A practitioner's Approach", Seventh Edition, McGraw Hill International Education, 2016.
2. Ian Sommerville, "Software Engineering", Sixth Edition, Pearson Education, (2001).

Reference Book(s):

1. Jim Arlow, Ila Neustadt, "UML 2 and the Unified Process: Practical Object-Oriented Analysis and Design", 2nd Edition, Pearson, (2005).
2. John W. Satzinger, Robert B. Jackson, Stephen D. Burd, Object-oriented analysis and design with the Unified process, Cengage Learning
3. James Rumbaugh, Ivar Jacobson, Grady Booch, The Unified modeling language Reference manual, Addison-Wesley

SEMESTER –V

NARAYANA ENGINEERING COLLEGE::GUDUR								
SEM-V	**ARTIFICIAL INTELLIGENCE**							**R21**
Course Code	**Hours / Week**			**Total hrs**	**Credit**	**Max Marks**		
	L	**T**	**P**		**C**	**CIE**	**SEE**	**TOTAL**
21CS2008	3	0	0	50	3	40	60	100

COURSE CONTENT		
MODULE – 1		**8H**
AI problems, foundation of AI and history of AI intelligent agents: Agents and Environments, the concept of rationality, the nature of environments, structure of agents, problem solving agents, problem formulation.		
MODULE – 2		**9H**
Problem Solving: Problem solving agents, toy problems, Real-world problems, searching for solutions. **Uninformed Search strategies**: BFS, DFS, Depth-limited search.		
MODULE – 3		**12H**
Informed Search strategies: GBFS, A* search, Local search algorithms: Hill-climbing. Adversarial Search: Games, optimal decision in games, Alpha-Beta pruning, Imperfect, Real-Time Decisions.		
MODULE – 4		**9H**
Knowledge and reasoning: Logical Agents: Knowledge -based Agents, The WUMPUS world, Logic, Propositional Logic, Reasoning Patterns in Propositional logic, Resolution, Forward and Backward chaining. First-order Logic: Syntax and Semantics of First-Order Logic		
MODULE – 5		**12H**
Learning and Knowledge formation in learning: Learning from Observations- Forms of Learning, Inductive Learning, Learning Decision Trees, and Ensemble Learning**.** A Logical formulation of learning, knowledge in learning, Explanation-Based Learning, Learning using Relevance Information **Expert systems:-** Introduction, basic concepts, structure of expert systems, the human element in expert systems how expert systems works		
	Total hours:	**50 hours**

TEXT BOOK:

1. Artificial Intelligence a Modern Approach, Stuart Russell, Peter Norvig (Person Education), 3rd edition.
2. Nils J. Nilsson, "Artificial Intelligence: A new Synthesis", Harcourt Asia Pvt. Ltd., 2000.

REFERENCES:

1. Artificial Intelligence- Rich E & Knight K (TMH), 4thedition.
2. Artificial Intelligence Structures and Strategies complex problem Solving – George F. Lugar Pearson Education.
3. D.W. Patterson, –Introduction to AI and Expert Systems‖, PHI, 1992...
4. R. J. Schalk off, ―Artificial Intelligence-an Engineering Approach‖, McGraw Hill Int. Ed., Singapore, 1992.

NARAYANA ENGINEERING COLLEGE::GUDURU								
SEM-V	**DESIGN AND ANALYSIS OF ALGORITHMS**							**R-21**
Course Code	**Hours / Week**			**Total hrs**	**Credit**	**Max Marks**		
	L	**T**	**P**		**C**	**CIE**	**SEE**	**TOTAL**
21CS2009	3	0	0	50	3	40	60	100
Pre-requisite: The basic knowledge of Data structure, Programming Language, Mathematics, and problem-solving ability.								

COURSE CONTENT		
MODULE – 1	**Introduction**	**10H**
Introduction: Algorithm, Algorithm specification, Performance analysis. Divide and Conquer: General method, Binary Search, Finding the maximum and minimum, Merge sort, Quick Sort, Selection, Strassen's matrix multiplication.		
MODULE -2	**Greedy Method and Dynamic Programming**	**10H**
Greedy Method: General method, Knapsack problem, Job Scheduling with Deadlines, Minimum cost Spanning Trees, Optimal storage on tapes, Single-source shortest paths. **Dynamic programming**: General Method, Multistage graphs, All-pairs shortest paths, Optimal binary search trees, 0/1 knapsack, the traveling salesperson problem.		
MODULE-3	**Traversal and Search Techniques and Back Tracking**	**10H**
Basic Traversal and Search Techniques: Techniques for binary trees, Techniques for Graphs, Network Topologies, Connected components and Spanning trees, Articulation point and Bi-connected components and DFS **Back tracking**: General Method, 8 – queens problem, Sum of subsets problem, Graph coloring and Hamiltonian cycles, Knapsack Problem.		
MODULE-4	**Branch and Bound and Lower Bound Theory**	**10H**
Branch and Bound: The method, Travelling salesperson, 0/1 Knapsack problem, Efficiency considerations. **Lower Bound Theory:** Comparison trees, Lower bounds through reductions – Multiplying triangular matrices, inverting a lower triangular matrix, computing the transitive closure.		
MODULE-5	**NP-Hard and NP-Complete Problems**	**10H**
NP – Hard and NP – Complete Problems: NP Hardness, NP Completeness, Consequences of being in P, Cook's Theorem, Reduction Source Problems, Reductions: Reductions for some known problems.		
	Total hours:	**50 hours**

Text Book(s):

1. Ellis Horowitz, Sartaj Sahni and S Rajasekaran, "Fundamentals of Computer Algorithms",2nd Edition,2012,University Press.
2. Jon-Kleinberg-Eva-Tardos, Algorithm Design, Pearson; 1st edition

Reference Book(s):

1. Anany Levitin, "Introduction to the Design and Analysis of Algorithms", Third Edition, Pearson Education,2012.
2. Thomas H.Cormen, Charles E.Leiserson, RonaldL. Rivestand Clifford Stein," Introduction to Algorithms", Third Edition, PHI Learning Private Limited, 2012.
3. Alfred V.Aho, John E.Hopcroft and Jeffrey D. Ullman,"Data Structures and Algorithms", Pearson Education,Reprint2006.

NARAYANAENGINEERINGCOLLEGE::GUDUR								
SEM-V	**THEORY OF COMPUTATION**							**R21**
Course Code	**Hours/Week**			**Total hrs**	**Credit**	**Max Marks**		
	L	**T**	**P**		**C**	**CIE**	**SEE**	**TOTAL**
21CS2010	3	0	0	50	3	40	60	100

COURSECONTENT		
MODULE– 1		**10H**
Finite Automata: Alphabets, Strings, Languages. Deterministic Finite Automata (DFA), Non Deterministic Finite Automata (NFA), Finite Automata with Epsilon transitions(ϵ-NFA or NFA-ϵ),Finite Automata with output, Conversion of on emachine to another, Minimization of Finite Automata.		
MODULE– 2		**10H**
Regular Expressions: Regular Expressions, Regular Sets, Identity Rules, Equivalence of two Regular Expressions, Manipulations of Regular Expressions, Finite Automata and Regular Expressions, Inter Conversion, Equivalence between Finite Automata and Regular Expressions, Pumping Lemma, Closure Properties, Applications of Regular Expressions, Finite Automata and Regular Grammars, Regular Expressions and Regular Grammars.		
MODULE– 3		**10H**
Context Free Grammars: Grammars, Chomsky Hierarchy, Context Free Grammar, Leftmost and Rightmost Derivations, Parse Trees, Ambiguous Grammars, Simplification of Context Free Grammars-Elimination of Useless Symbols, EProductions and Unit Productions, Normal Forms for Context Free Grammars-Chomsky Normal Form and Greibach Normal Form, Closure Properties, Applications of Context Free Grammars.		
MODULE– 4		**10H**
Push Down Automata: Push down Automata, Definition, Instantaneous Description Language Acceptance of pushdown Automata, Design of Pushdown Automata, Deterministic and Non–Deterministic Push down Automata, Equivalence of Push down Automata and Context Free Grammars Conversion, Application of Push down Automata.		
MODULE– 5		**10H**
Turing Machines and Undecidability: Basics of Turing Machine (TM), Transitional Representation of TMs, Instantaneous description, Non Deterministic TM, Conversion of Regular Expression to TM, Types of Turing machines (proofs not required), Linear bounded Automata and Context sensitive language, Unrestricted grammar, Properties of Recursive and Recursively enumerable languages.		
	Total hours:	**50 hours**

TEXTBOOK:

1. J.E.Hopcroft, R.Motwaniand J.D. Ullman, Introduction to Automata Theory, Languages and Computation, 3rd Edition, Pearson, 2008.
2. Michael Sipser, Introduction to the Theory of Computation, Second Edition, Thomson Course Technology

REFERENCES:

1. Formal Language and Automata Theory, K.V.N.SunithaandN.Kalyani,Pearson,2015.
2. Introduction to Automata Theory, Formal Languages and Computation, Shyamalendu Kandar, Pearson, 2013.

NARAYANA ENGINEERING COLLEGE:: GUDUR								
SEM-V	**ARTIFICIAL INTELLIGENCE LABORATORY**							**R21**
Course Code	**Hours / Week**			**Total hrs**	**Credit**	**Max Marks**		
	L	**T**	**P**		**C**	**CIE**	**SEE**	**TOTAL**
21CS2505	0	0	2	36	1	40	60	100

List of Experiments	
TASK – 1	**3H**
Implementation of DFS and BFS	
TASK – 2	**3H**
Implementation of travelling salesman Problem	
TASK – 3	**3H**
Implementation of simple Chabot.	
TASK – 4	**3H**
Implementation of wampus world problem.	
TASK – 5	**3H**
Implementation of 8 puzzle problem	
TASK – 6	**3H**
Implementation of Towers of Hanoi problem	
TASK – 7	**3H**
Implementation of A* Algorithm	
TASK – 8	**3H**
Implementation of Hill Climbing Algorithm	
TASK – 9	**3H**
Implementation of Simulated Annealing Algorithm.	
TASK – 10	**3H**
Implementation of Knowledge representation schemes.	
TASK – 11	**3H**
Study of PROLOG programming language and its functions. Demonstrate knowledge representation for the following using prolog a. Ram likes mango. b. Seema is a girl. c. Bill likes Cindy. d. Rose is red. e. John owns gold	
TASK – 12	**3H**
Implementation of any case study using AI techniques	
Total hours:	**36 hours**

TEXT BOOK:

1. Artificial Intelligence, 2nd Edition, E. Rich and K. Knight, TMH.
2.Artificial Intelligence a Modern Approach, Stuart Russell, Peter Norvig (Person Education), 3nd edition.

REFERENCES:

1. Python Essential Reference, David M. Beazley, Pearson Education, Inc.
2. Fluent Python, Luciano Ramalho by O'Reilly Media
3. Python Cookbook, David Beazley and Brian K. Jones, O'Reilly Atlas.3e
4. Artificial Intelligence- Rich E & Knight K (TMH), 4th edition.
5. Artificial Intelligence Structures and Strategies complex problem Solving – George F. Lugar Pearson Education.

NARAYANA ENGINEERING COLLEGE:: GUDUR								
SEM-V	**CODING LAB**							**R21**
Course Code	**Hours / Week**			**Total hrs**	**Credit**	**Max Marks**		
	L	**T**	**P**		**C**	**CIE**	**SEE**	**TOTAL**
21CS2506	0	0	2	39	1	40	60	100

List of Experiments

TASK 1: MAXIMUM NUMBER OF GUESTS — 2H

Problem Statement

A party has been organised on cruise. The party is organised for a limited time (T). The number of guests entering (E[i]) and leaving (L[i]) the party at every hour is represented as elements of the array. The task is to find the maximum number of guests present on the cruise at any given instance within T hours.

Example 1:
Input :

- 5 -> Value of T
- [7,0,5,1,3] -> E[], Element of E[0] to E[N-1], where input each element is separated by new line
- [1,2,1,3,4] -> L[], Element of L[0] to L[N-1], while input each element is separate by new line.

Output:
8 -> Maximum number of guests on cruise at an instance.

TASK 2: BUNCH OF BALLOONS — 2H

Problem Statement

At a fun fair, a street vendor is selling different colours of balloons. He sells N number of different colours of balloons (B[]). The task is to find the colour (odd) of the balloon which is present odd number of times in the bunch of balloons.

Note: If there is more than one colour which is odd in number, then the first colour in the array which is present odd number of times is displayed. The colours of the balloons can all be either upper case or lower case in the array. If all the inputs are even in number, display the message “All are even”.

Example 1:

- 7 -> Value of N
- [r,g,b,b,g,y,y] -> B[] Elements B[0] to B[N-1], where each input element is separated by new line.

Output :

- r -> [r,g,b,b,g,y,y] -> “r” colour balloon is present odd number of times in the bunch.

Explanation:
From the input array above:

- r: 1 balloon
- g: 2 balloons
- b: 2 balloons
- y : 2 balloons

Hence , r is only the balloon which is odd in number.

TASK 3: FOREST FIRE	2H

Problem Description

Roco is an island near Africa which is very prone to forest fire. Forest fire is such that it destroys the complete forest. Not a single tree is left.This island has been cursed by God , and the curse is that whenever a tree catches fire, it passes the fire to all its adjacent tree in all 8 directions, North, South, East, West, North-East, North-West, South-East, and South-West. And it is given that the fire is spreading every minute in the given manner, i.e every tree is passing fire to its adjacent tree. Suppose that the forest layout is as follows where T denotes tree and W denotes water.

Your task is that given the location of the first tree that catches fire, determine how long would it take for the entire forest to be on fire. You may assume that the layout of the forest is such that the whole forest will catch fire for sure and that there will be at least one tree in the forest

Input Format:

- First line contains two integers, M, N, space separated, giving the size of the forest in terms of the number of rows and columns respectively.
- The next line contains two integers X,Y, space separated, giving the coordinates of the first tree that catches the fire.
- The next M lines, where i^{th} line containing N characters each of which is either T or W, giving the position of the Tree and Water in the ith row of the forest.

Output Format:

Single integer indicating the number of minutes taken for the entire forest to catch fire

Constrains:

- $3 \leq M \leq 20$
- $3 \leq N \leq 20$

Sample Input 1:

3 3
W T T
T W W
W T T

Sample Output 1:

5

Explanation:

In the second minute,tree at (1,2) catches fire,in the third minute,the tree at (2,1) catches fire,fourth minute tree at (3,2) catches fire and in the fifth minute the last tree at (3,3) catches fire.

Sample Input 2:

6 6
1 6
W T T T T T
T W W W W W
W T T T T T
W W W W W T
T T T T T T
T W W W W W

Sample Output 2:

16

TASK 4: RAILWAY STATION	2H

Problem Description
Given schedule of trains and their stoppage time at a Railway Station, find minimum number of platforms needed.
Note -
If Train A's departure time is x and Train B's arrival time is x, then we can't accommodate Train B on the same platform as Train A.

Constraints
1 <= N <= 10^5
0 <= a <= 86400
0 < b <= 86400
Number of platforms > 0

Input
First line contains N denoting number of trains.
Next N line contain 2 integers, a and b, denoting the arrival time and stoppage time of train.

Output
Single integer denoting the minimum numbers of platforms needed to accommodate every train.

Example 1
Input
3
10 2
5 10
13 5

Output
2

Explanation
The earliest arriving train at time t = 5 will arrive at platform# 1. Since it will stay there till t = 15, train arriving at time t = 10 will arrive at platform# 2. Since it will depart at time t = 12, train arriving at time t = 13 will arrive at platform# 2.

Example 2
Input
2
2 4
6 2

Output
2

Explanation
Platform #1 can accommodate train 1.
Platform #2 can accommodate train 2.
Note that the departure of train 1 is same as arrival of train 2, i.e. 6, and thus we need a separate platform to accommodate train 2.

TASK 5: COUNTING ROCK SAMPLES	1H

Juan Marquinho is a geologist and he needs to count rock samples in order to send it to a chemical laboratory. He has a problem: The laboratory only accepts rock samples by a range of its size in ppm (parts per million).

Juan Marquinho receives the rock samples one by one and he classifies the rock samples according to the

range of the laboratory. This process is very hard because the number of rock samples may be in millions.

Juan Marquinho needs your help, your task is to develop a program to get the number of rocks in each of the ranges accepted by the laboratory.

Input Format:

An positive integer S (the number of rock samples) separated by a blank space, and a positive integer R (the number of ranges of the laboratory); A list of the sizes of S samples (in ppm), as positive integers separated by space R lines where the ith line containing two positive integers, space separated, indicating the minimum size and maximum size respectively of the ith range.

Output Format:

R lines where the ith line containing a single non-negative integer indicating the number of the samples which lie in the ith range.

Constraints: 10 ? S ? 10000 1 ? R ? 1000000 1?size of each sample (in ppm) ? 1000

Example 1

Input: 10 2
345 604 321 433 704 470 808 718 517 811
300 350
400 700
Output: 2 4

Explanation:

There are 10 samples (S) and 2 ranges (R). The samples are 345, 604,811. The ranges are 300-350 and 400-700. There are 2 samples in the first range (345 and 321) and 4 samples in the second range (604, 433, 470, 517). Hence the two lines of the output are 2 and 4

Example 2

Input: 20 3
921 107 270 631 926 543 589 520 595 93 873 424 759 537 458 614 725 842 575 195
1 100
50 600
1 1000

Output: 1 12 20
Explanation:

There are 20 samples and 3 ranges. The samples are 921, 107 195. The ranges are 1-100, 50-600 and 1-1000. Note that the ranges are overlapping. The number of samples in each of the three ranges are 1, 12 and 20 respectively. Hence the three lines of the output are 1, 12 and 20.

TASK 6: SORTING BOXES	**2H**

Problem Statement

The parcel section of the Head Post Office is in a mess. The parcels that need to be loaded to the vans have been lined up in a row in an arbitrary order of weights. The Head Post Master wants them to be sorted in the increasing order of the weights of the parcels, with one exception. He wants the heaviest (and presumably the most valuable) parcel kept nearest his office.

You and your friend try to sort these boxes and you decide to sort them by interchanging two boxes at a time. Such an interchange needs effort equal to the product of the weights of the two boxes.

The objective is to reposition the boxes as required with minimum effort.

Input Format:

- The first line consists of two space-separated positive integers giving the number of boxes (N) and the position of the Head Post Masters office (k) where the heaviest box must be.
- The second line consists of N space-separated positive integers giving the weights of the boxes. You may assume that no two weights are equal

Output Format:

- The output is one line giving the total effort taken to get the boxes in sorted order, and the heaviest in position k.

Constraints:

N<=50 and Weights <= 1000

Sample Input 1:

5 2

20 50 30 80 70

Sample Output 1:

3600

TASK 7: CONSTELLATION	**3H**

Description:

Three characters {#, *, . } represents a constellation of stars and galaxies in space. Each galaxy is demarcated by # characters. There can be one or many stars in a given galaxy. Stars can only be in the shape of vowels {A, E, I, O, U}. A collection of * in the shape of the vowels is a star. A star is contained in a 3×3 block. Stars cannot be overlapping. The dot (.) character denotes empty space.

Given a matrix of dimensions 3×N consisting of **{#, *, .}** character, the task is to find the galaxy and stars within them.

NOTE: Please pay attention to how vowels A is denoted in 3x3 block in the examples section below.

Example 1:

Input: *18*

*. *# * * * # * * * # * * *. *.

*. *# *. *#. *. # * * * * * *

* * *# * * *# * * *# * * * *. ***Output:***U#O#I#EA*

Explanation:

It can be seen that the stars make the image of the alphabets U, O, I, E and A respectively.

Example 2:

Input:*N = 12*

.#.***#.*.

.#..*.#***

#.#*.*

Output:

U#I#A

TASK 8: RITIK - PRIME TIME AGAIN	1H

Description:

Here on earth, our 24-hour day is composed of two parts, each of 12hours. Each hour in each part has a corresponding hour in the other part separated by 12 hours: the hour essentially measures the duration since the start of the day part. For example, 1 hour in the first part of the day is equivalent to 13, which is 1 hour into the second part of the day. Now, consider the equivalent hours that are both prime numbers.
We have 3 such instances for a 24-hour 2-part day:

- 5~17
- 7~19
- 11~23

Accept two natural numbers D, P >1 corresponding respectively to numberof hours per day and numberof parts in a day separated by a space. D should be divisible by P, meaning that the number of hours perpart (D/P) should be a natural number. Calculate the number of instances of equivalent prime hours.
Output zero if there is no such instance.
Note That we require each equivalent hour in each part in a day to be a prime number.

Example:

Input:

24 2

Output:

3 (We have 3 instances of equivalent prime hours: 5~17, 7~19 and 11~23.)

Constraints:

10 <= D < 500

2 <= P < 50

Input:

Single line consists of two space separated integers, D and P corresponding to number of hours per dayand number of parts in a day respectively.

Output: Output must be a single number, corresponding to the number of instances of equivalentprime number, as described above

Example Input

36 3

Output

2

Explanation

In the given test case D = 36 and P = 3

Duration of each day part = 12 2~14~X

3~15~X

5~17~29 – instance of equivalent prime hours

7~19~31 – instance of equivalent prime hours

11~23~X

Hence the answers is 2.

TASK 9: MINIMUM GIFTS	3H

Description:

A company has decided to offer all of its staff employees. For that, the company has given some rank to each Employee. Based on that rank, the company has made certain rules to distribute the gifts. The rules for distributing the gifts are: Each employee must receive at least one gift. Employees having higher ranking get a greater number of gifts than their neighbors. What is the minimum number of gifts required by the company?

Constraints:

$1 < T < 10$

$1 < N < 100000$

$1 < Rank < 10^9$

Input

The First line contains integer T, denoting the number of test cases.

For each test case: First-line contains integer N, denoting the number of employees.

The Second-line contains N space-separated integers, denoting the rank of each employee.

Output

For every test case, print the number of minimum gifts required by the company on a new line.

Now let us understand the statement with the help of an example.

TASK 10: MINIMIZING TH SUM PROBLEM	3H

A well know property builder Vatika was building a new residential project in the prime location of National Highway of Gurgaon. Various families invested in this project and purchased flats. Some people

also bought more than one flat in the society. Along with flat, families also bought same parking space with their flats but the payment will be done at the time of possession only.

At the possession time of project, builder started allocating the parking lots alloted to each of the flat member. Allotment of parking space is done according to number of flats bought by the family. Builder decided to allot parking space to the investor in the form of rectangular plots inside parking area of different sizes with different cost per unit area. So, he sold these parking spaces to the flat owners but made a mistake. He made partitions that could be overlapping. When the investors came to know about it, they ran to him for compensation of extra money they paid to him. So, he decided to return all the money to the investors of that parking space which was overlapping with other investor's parking space to settle down the conflict. All the portion of conflicted area will be taken back by the builder.

To decide the total compensation, he has to calculate the total amount of money to return back to investors with the same cost they had purchased from him. Suppose, Builder has a total parking area of 1000 x 1000 equal square blocks where each block is equivalent to a unit square area which can be represented on the co-ordinate axis. Now find the total amount of money, he has to return to the investors. Help Builder to accomplish this work so that there will be no conflict between investors.

Input Format

(X1, Y1) and (X2, Y2) are the locations of first and last square block on the diagonal of the rectangular region.

Input 1: It will be the integer type which shows the total number of parking spaces N that builder distributed.

Input 2: Second input will be multi-dimension array where

Next line will be the number of rows and each row is having 5 values like (X1, Y1), (X2, Y2) to represent a rectangular area of land, and cost per unit area C.

Constraints

1 <= N <= 100

-10^4 <= X, Y <= 10^4

Output Format

It will the integer which shows the total amount builder has to return to investors to resolve this conflict.

Sample Test Case 1

Input

3
3
5
1 4 4 6 1
4 3 6 6 2
2 2 5 4 3

Output

35

Explanation:

For given sample input (see given graph for reference), compensation money for different investors is as follows:

Investor with land area A: C1 = 5 * 1 = 5

Investor with land area B: C2 = 6 * 2 = 12

Investor with land area C: C3 = 6 * 3 = 18

Total Compensation Money = C1 + C2 + C3 = 5 + 12 + 18 = 35

TASK 11: LAMBDA PRIME	3H

Description: Given an array of integers, perform at most K operations so that the sum of elements of final array is minimum. An operation is defined as follows – Consider any 1 element from the array[i]. Replace arr[i] by floor(arr[i]/2). Perform next operations on the updated array. The task is to minimize the sum after utmost k operations.

Constraints:

1<=N, K<= 10^5.

Input:

First line contains two integers N and K representing size of array and maximum number of operations that can be performed

On the array respectively.

Second line contains N space separated integers denoting the elements of the array,arr.

Output:

Print a single integer denoting the minimum sum of final array.
Examples:

Input:

4 3

20 7 5 4
Output:
17

TASK 12: CRITICAL PLANETS	**3H**

Description: The war between Republic and Separatists is escalating. The Separatists are on a new offensive. They have started blocking the path between the republic planets (represented by integers) so that these planets surrender due to the shortage of food and supplies. The Jedi council has taken note of the situation and they have assigned Jedi Knight Skywalker and his Padawan Ahsoka to save the critical planets from blockade (Those planets or system of planets which can be accessed by only one path and may be lost if that path is blocked by separatist).
Skywalker is preparing with the clone army to defend the critical paths. He has assigned Ahsoka to find the critical planets. Help Ahsoka to find the critical planets(C) in ascending order. You only need to specify those planets which have only one path between them and they cannot be accessed by any other alternative path if the only path is compromised.

Constraints

M <= 10000
N <= 7000

Input

First line contains two space separated integers M and N, where M denotes the number of paths between planets and N denotes the number of planets.
Next M lines, each contains two space separated integers, representing the planet numbers that have a path between them.

Output

C lines containing one integer representing the critical planet that they need to save in ascending order of the planet number if no planet is critical then print -1

TASK 13: COLLECTING CANDIES	**3H**

Krishna loves candies a lot, so whenever he gets them, he stores them so that he can eat them later whenever he wants to.

He has recently received N boxes of candies each containing Ci candies where Ci represents the total

number of candies in the i th box. Krishna wants to store them in a single box. The only constraint is thathe can choose any two boxes and store their joint contents in an empty box only. Assume that therearean infinite number of empty boxes available.

At a time, he can pick up any two boxes for transferring and if both the boxes contain X and Y number ofcandies respectively, then it takes him exactly X+Y seconds of time. As he is too eager to collect all of them he has approached you to tell him the minimum time in which all the candies can be collected.

Input Format:

- The first line of input is the number of test case T
- Each test case is comprised of two inputs
- The first input of a test case is the number of boxes N
- The second input is N integers delimited by whitespace denoting the number of candies in eachbox

Output Format: Print minimum time required, in seconds, for each of the test cases. Print each outputon a new line.

Constraints:

- $1 < T < 10$
- $1 < N < 10000$

$1 <$ [Candies in each box] < 100009

TASK 14: COUNT CROSSING BIKER'S	3H

Description:
Given an array Arr[] of size T, contains binary digits, where

- 0 represents a biker running to the north.
- 1 represents a biker running to the south.

The task is to count crossing biker in such a way that each pair of crossing biker(N, S), where 0<=N<S<T, is passing when N is running to the north and S is running to the south.

Constraints:

0<=N<S<T

Example 1:

Input :

5 -> Number of elements i.e. T

0 -> Value of 1st element.

1 -> Value of 2nd element

0 -> Value of 3rd element.

1 -> Value of 4th element.

1 -> Value of 5th element

Output :

5

TASK 15: STRING SLICE	2H

You need to take string input and two other numbers which will be the start and end point of the slice and you need to print that slice of string.

Input Format

You will be given a function with string and two other integers as arguments.

Constraints

1 <= |S| <= 10^3

Output Format

You need to return the slice of the string.

Sample Test Case 1

Input

Hello Techgig

1

4

Output

Ello

TASK 16: MONKEYS IN THE GARDEN	2H

In a garden, trees are arranged in a circular fashion with an equal distance between two adjacent trees. The height of trees may vary. Two monkeys live in that garden and they were very close to each other. One day they quarrelled due to some misunderstanding. None of them were ready to leave the garden. But each one of them wants that if the other wants to meet him, it should take maximum possible time to reach him, given that they both live in the same garden. The conditions are that a monkey cannot directly jump from one tree to another. There are 30 trees in the garden. If the height of a tree is H, a monkey can live at any height from 0 to H. Let's say he lives at the height of K then it would take him K unit of time to climb down to the ground level. Similarly, if a monkey wants to climb up to K height it would again take K unit of time. The time to travel between two adjacent trees is 1 unit. A monkey can only travel in a circular fashion in the garden because there is a pond at the center of the garden.

So the question is where two monkeys should live such that the travelling time between them is maximum while choosing the shortest path between them in any direction clockwise or anti-clockwise. You have to answer only the maximum travelling time.

Input Format

The first line consists of total number of trees (N)

Each of the following N lines contains the height of trees in a clockwise fashion.

Constraints

1 <= Total Trees <= 30

1 <= Height Of Trees(H) <= 10000

Output Format

You must print an integer which will be the maximum possible travel time.

Sample Test Case 1

Input

4

1

2

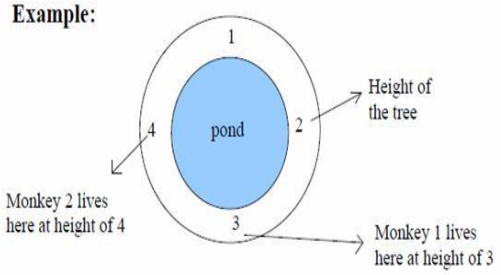
3

4

Output

8

Example:



In above example if monkey 1 wants to meet monkey 2 it would take total time :
3(climb down the tree) + 1(time between the tree)+ 4(climb up the tree)= 8 unit .

TASK 17: RITIK - THE DAY DREAMER	1H

Ritik is a man of dreams, he keeps dreaming day and night, wondering about space and time, sci-fi and a lot more. Today is yet another day when he dreamed about a Galactic Grid where he was supposed to rescue his friend Kriti. Now this Grid is slightly unusual consisting of two species Vilgax and Arkaknight.
Vilgax is bad and can freeze you for some time whereas Arkaknight is good and lets you jump from one place to another.
The Grid is made up of m rows and n columns. Each cell consists of an integer either positive, negative or zero. There are exactly two zeroes one at top-left corner and the other at bottom-right corner of the grid representing initial positions of Ritik and Kriti respectively.
A positive integer represents an Arkaknight. It gives you the power to jump from cell to another containing the same integer anywhere within the grid. For example - a cell containing 2 can let you jump to any other cell containing 2 in the grid if there is one.
A negative integer represents a Vilgax. It freezes for a certain period of time that is you cannot move for certain units of time represented by the absolute value of the integer present in that cell. For example , a cell containing -3 makes you freeze for 3 units of time in that cell i.e. you cannot move or jump to any other cell.
Ritik can also move to any cell that share a common wall i.e. adjacent to each other (top, left, right and bottom) within the grid. It takes one unit of time to jump from one cell to another.
Your task is to tell the minimum units of time taken by Ritik to reach Kriti given the grid of m x n.

Input Format

First line of input contains number of rows - M.
Second line of input contains number of columns - N.
Then M lines of input follows each containing N integers respectively.
Note: The Positive integers will range from [1, M * N] both inclusive and negative integers can range from [-1 , -(M * N)] both inclusive.

Constraints

1 <= M <= 1000
1 <= N <= 1000

Output Format

Print the minimum units of time taken by Ritik to reach Kriti.

Sample Test Case 1

Input

3
3
0 2 3
-1 2 -3
3 2 0

Output

3

Explanation

From initial position of Ritik, he can move to right cell or bottom cell, moving to bottom cell containing -1 can cause him to freeze for 1 unit of time so he will move to cell containing 2, now from here he can to any of the 2 in the grid or move to adjacent cell, best is to move to cell (3, 2) containing 2, now from here he can move to cell (3, 3) where Kriti is present. Hence 3 moves.

TASK 18: CHECK DIVISIBILITY	1H

You have to divide a by b. If there is any arithmetic exception than print it else print the correct output.

Input Format

The only line of input consist of a and b.

Constraints

NA

Output Format

Print the exception if any else prints the correct output.

Sample Test Case 1

Input

5 0

Output

java.lang.ArithmeticException: / by zero

Total hours:	**39 hours**

TEXT BOOK:

1. Artificial Intelligence, 2nd Edition, E. Rich and K. Knight, TMH.
2. Artificial Intelligence a Modern Approach, Stuart Russell, Peter Norvig (Person Education), 3nd edition.

REFERENCES:

1. Python Essential Reference, David M. Beazley, Pearson Education, Inc.
2. Fluent Python, Luciano Ramalho by O'Reilly Media
3. Python Cookbook, David Beazley and Brian K. Jones, O'Reilly Atlas.3e
4. Artificial Intelligence- Rich E & Knight K (TMH), 4th edition.
5. Artificial Intelligence Structures and Strategies complex problem Solving – George F. Lugar Pearson Education.

NARAYANA ENGINEERING COLLEGE::NELLORE								
SEM-V	**DESIGN AND ANALYSIS OF ALGORITHMS LABORATORY**							**R21**
Course Code	**Hours / Week**			**Total hrs**	**Credit**	**Max Marks**		
	L	**T**	**P**		**C**	**CIE**	**SEE**	**TOTAL**
21CS2507	0	0	2	36	1	40	60	100

List of Experiments	
TASK – 1	**3H**
1. a) Implementation of Binary search algorithm. b) Implementation of Binary search algorithm using Divide & Conquer method.	
TASK – 2	**3H**
2. a) Implementation of Quick Sort algorithm. b) Implementation of Quick Sort algorithm using Divide & Conquer method.	
TASK – 3	**3H**
3. a) Program to merge two sorted arrays. b) Implementation of Merge Sort algorithm using Divide & Conquer method	
TASK – 4	**3H**
.4. a) Implementation of Matrix multiplication. b) Implementation of Strassen's Matrix multiplication	
TASK – 5	**3H**
5. a) Program to implement knapsack problem using greedy method. b) Program to implement job sequencing with deadlines using greedy method.	
TASK – 6	**3H**
6. a) Find Minimum Cost Spanning Tree of a given undirected graph using Kruskal's algorithm. b) Find Minimum Cost Spanning Tree of a given undirected graph using Prim's algorithm.	
TASK – 7	**3H**
7. a) Print all the nodes reachable from a given starting node in a digraph using BFS method. b) Check whether a given graph is connected or not using DFS method.	
TASK – 8	**3H**
8. a) Implementation of Optimal merge patterns. b) Implement travelling salesman problem.	
TASK – 9	**6H**
9 .a) Program for finding shortest path for multistage graph using dynamic programming. b) Implement 0/1 Knapsack problem using Dynamic Programming.	
TASK – 10	**3H**
10 Program to implement 8-queens problem using backtrack method.	
ADDITIONAL EXPERIMENTS	**6H**
1. Implement All-Pairs Shortest Paths Problem using Floyd's algorithm. 2. Find a subset of a given set S = {s1,s2,.....,sn} of n positive integers whose sum is equal to a given positive integer d. For example, if S= {1, 2, 5, 6, 8} and d = 9 there are two solutions {1,2,6} and {1,8}.A suitable message is to be displayed if the given problem instance doesn't have a solution.	
Total hours:	**36 hours**

SEMESTER –VI

NARAYANAENGINEERINGCOLLEGE::GUDUR								
SEM-VI	**MACHINE LEARNING**							**R21**
Course Code	**Hours/Week**			**Total hrs**	**Credit**	**Max Marks**		
	L	**T**	**P**		**C**	**CIE**	**SEE**	**TOTAL**
21CS2011	3	0	0	48	3	40	60	100

Pre-requisite: Familiarity with basic concepts of computer science(algorithms , data structures and complexity) mathematical maturity

Course Objectives:
1. Gain knowledge about basic concepts of Machine Learning
2. Study about different learning algorithms
3. Learn about Artificial Neural Network learning strategies
4. Familiar with Regression concepts
5. Study about instance based learning and reinforcement learning

Course Outcomes: After successful completion of the course, Student will be able to:

CO 1	Understand the concepts of computational intelligence like machine learning
CO 2	Understand and apply the various Machine learning strategies
CO 3	Familiar with basic concepts in artificial neural network and its learning methods
CO 4	Explore regression methods in machine learning
CO 5	Design and analyze the instance based and reinforcement learning

CO-PO Mapping

CO	**PO**												**PSO**	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	3	2	3	3								2	2	3
CO2	2	3	2	2	2	3						3	2	3
CO3	2	3	3	3								3	2	3
CO4	3	2			2	3						3	3	3
CO5	3	2	2	3	3	3						2	2	2
1-Low, 2- Medium, 3- High														

COURSE CONTENT		
MODULE – 1	**Introduction**	**8H**
Introduction: Learning – Types of Machine Learning – Supervised Learning, Relationship between ML and human learning, Example applications of ML-Designing a Learning system, Perspective and Issues in Machine Learning. Concept Learning Task – Concept Learning as Search – Finding a Maximally Specific Hypothesis – Version Spaces and the Candidate Elimination Algorithm.		
At the end of the Module 1, students will be able to: 1. Understandbasicconceptsofmachinelearning(Ll) 2. Compare machine learning and human learning(L2) 3.Analyzemachinelearningtechniques(L4)		
MODULE -2	**CLASSIFICATION**	**9 H**
Classification: Supervised Learning-The problem of classification-Training and testing classifier models-Decision Tree-Naive Bayes classification-Bayesian networks--Ensemble Learning-Support Vector Machines-Cross-validation-Model evaluation (precision, recall, F1-mesure, accuracy)-Applications of classifications.		
At the end of the Module 2, students will be able to: 1. Differentiate supervised and unsupervised learning methods (IA). 2.Solve classification problem using k-nearest neighbor classifier (L3). 3.Apply Naïve Bayes classifier to solve decision making problem (L3).		
MODULE-3	**ANN and Regression**	**11 H**
Artificial Neural Networks: Introduction, Neural Network representation, Appropriate problems, Perceptrons, Multilayer networks and Back propagation algorithm. **Regression**: Linear Regression-Multi-variable regression-Model evaluation-Least squares regression-Logistic regression -Gradient Descent Algorithm-Applications of regression		
At the end of the Module 3, students will be able to: 1. Determine Clusters in data using k-means and Hierarchical Clustering methods(L5). **2.**Reinember applications of clustering techniques		
MODULE-4	**UNSUPERVISED LEARNING**	**9 H**
Unsupervised Learning – K-Means Algorithm-Hierarchical and density based Clustering-Applications of Clustering.		
At the end of the Module 4, students will be able to: 1. Describe gradient descent approach, maximum likelihood estimation and method of least squares (LI). 2.Apply SVM to determine a hyper plane with maximum margin(L3).		
MODULE-5	**INSTANCE AND REINFORCEMENT LEARNING**	**11 H**
Instance Based Learning: Introduction, k-nearest neighbour learning, locally weighted regression, radial basis function, cased-based reasoning. **Reinforcement Learning:** Introduction, Learning Task, Q Learning; Introduction to Agents, Intelligent Agents – Problem Solving – Searching, Logical Agents.		
At the end of the Module 5, students will be able to: Understandinstantbasedlearningtechniques(L2)Under standreinforcementlearningtechniques(L2)		
	Total hours:	**48 HOURS**
Content Beyond Syllabus: Life cycle of ML		
Text Book(s): Tom M. Mitchell, Machine Learning, India Edition 2013, McGraw Hill Education.		

Reference Books :

1. Trevor Hastie, Robert Tibshirani, Jerome Friedman, The Elements of Statistical Learning, 2nd edition, springer series in statistics.
2. Ethem Alpaydın, Introduction to machine learning, second edition, MIT press

NARAYANAENGINEERINGCOLLEGE::GUDUR								
SEM-VI	**WEB TECHNOLOGIES**							**R21**
Course Code	**Hours/Week**			**Total hrs**	**Credit**	**Max Marks**		
	L	**T**	**P**		**C**	**CIE**	**SEE**	**TOTAL**
21CS2012	3	0	0	48	3	40	60	100

Pre-requisite: HTML, CSS, and JavaScript

Course Objectives:

On completion of this course, a student will be familiar with client server architecture and able to develop a web application using java technologies. Students will gain the skills and project-based experience needed for entry into web application and development careers.

Course Outcomes: After successful completion of the course, Student will be able to:

CO 1	Construct static web pages using HTML and CSS. (BL-3)
CO 2	Implement various concepts related to dynamic web pages and validate them using JavaScript. (BL-3)
CO 3	Explore the role of a web server in serving PHP pages. (BL-3)
CO 4	Develop web Applications using Scripting Languages. (BL-3)
CO 5	Understand the syntax of common SQL commands, including SELECT, INSERT, UPDATE, DELETE. (BL-3)

CO-PO Mapping

CO	**PO**												**PSO**	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	3	2			1								3	3
CO2	2	3	3		2								3	3
CO3	1	3	3		3								3	3
CO4	1	2	3	1	2								3	3
CO5	3	3	2		3								3	3

1-Low, 2- Medium, 3- High

COURSE CONTENT		
MODULE -1	**HTML, CSS**	**10H**
HTML: Introduction to Web, Basic Syntax, Standard HTML Document Structure, Basic Text Mark-up, HTML styles, Elements, Attributes, Heading, Layouts, HTML media, I frames Images, Hypertext, Links, Lists, Tables, Forms, Tage related to Forms creation, GET and POST method. **Cascading Style Sheets:** Measurement Units, CSS concepts related to: Colors, Backgrounds, Fonts, Text, Links, Tables, Borders, Margins, Lists, Padding, Cursors, Outlines, Dimension, Scrollbars.		
MODULE -2	**JavaScript**	**10H**
JavaScript: Introduction to JavaScript, Objects, Primitives Operations and Expressions, Control Statements, Arrays, Functions, Constructors, Regular Expressions, Exception Handling, Validation, Built-in objects, Event Handling, DHTML with JavaScript, DOM Model.		
MODULE -3	**Introduction to PHP**	**8H**
Introduction to PHP: Static vs Dynamic web Pages, Web Server, Configuring PHP, Anatomy of a PHP Page, Variables and data types, Operators, Expressions and Statements, Strings, Arrays and Functions.		
MODULE -4	**PHP Advanced Concepts**	**10H**
PHP Advanced Concepts: Using Cookies, Using HTTP Headers, Using Sessions, Authenticating users, Using Environment and Configuration variables, PHP Forms		
MODULE -5	**MySQL with PHP**	**10H**
MySQL with PHP : MySQL Introduction, Basic queries, PHP & MySQL - Environment Setup, Create Database, Drop Database, Select Database, Create Tables, Drop Tables, Insert Records, Select Records, Update Records, Delete Records, Where Clause, Like Clause, Sorting Data Using Joins. **Case Studies** *(HTML, CSS, PHP with MySQL) CRUD Operations* : (1) Student Result Entry and Viewing, (2) Core Banking Operations. (3) Any other Similar Cases which Involves CRUD operation.		
	Total Hours :	**48 Hours**

TEXT BOOK(S):

1. Robet WSebesta, **Programming the World Wide Web**,7th Edition, Pearson, 2013
2. UttamKRoy, **WebTechnologies**, 1stEdition,7th impression, Oxford, 2012
3. LeeBabin, Nathan A Good, Frank M. Kromann and Jon Stephens, **PHP 5 Recipes A problem Solution approach**.

REFERENCES:

1. Deitel and Deitel and Nieto, Internet and World Wide Web- How to Program, 5th Edition, Prentice Hall, 2011.
2. ELad Elrom, Pro Mean Stack Development,1st Edition, Apress O'Reilly, 2016
3. David sawyer mcfarl and JavaScript & jQuery the missing manual, 2nd Edition, O'Reilly, 2011
4. Peter Pollock, Web Hosting for Dummies, 1st Edition, & Sons, 2013
5. Tom Christiansen, Jonathan want, Programming Perl, 4th Edition, O'Reilly, 2012
6. KogentLS, WebTechnologies: HTML, JavaScript, PHP, Java, JSP, XML and AJAX, Black book, 1st Edition, DreamTech,2009
7. Paul Wang, Sanda SKatila, An Introduction to Web Design, Programming, 1st Edition, Cengage Learning, 2003

NARAYANAENGINEERINGCOLLEGE::GUDUR								
SEM-VII	**MACHINE LEARNING LABORATORY**							**R21**
Course Code	**Hours/Week**			**Total hrs**	**Credit**	**Max Marks**		
	L	**T**	**P**		**C**	**CIE**	**SEE**	**TOTAL**
21CS2508	0	0	2	36	1	40	60	100

Course Outcomes: After successful completion of the course, Student will be able to:	
CO 1	Introduction to Python and Python Libraries- NumPy, Pandas, Matplotlib, Scikit.
CO 2	Perform Data exploration and pre-processing in Python and Feature Engineering Feature Selection Methods.
CO 3	Implement and demonstrate the FIND-S algorithm for finding the most special hypothesis based on a given set of training data samples. Read the training data from .CSV file
CO 4	working of the decision tree based ID3 algorithm
CO 5	demonstrate the diagnosis of heart patients using standard Heart Disease Data

CO-PO Mapping														
CO	**PO**												**PSO**	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	3	2	3	2	2	2						2	2	3
CO2	3	2	3	2								2	3	3
CO3	3	2	3	2								3	2	2
CO4	2	2	2	3	2	2						3	2	2
CO5	3	2	3	2		2	3	2				3	2	2
1-Low, 2-Medium, 3- High														

List of Experiments	
TASK– 1	**3H**
Introduction to Python and Python Libraries- NumPy, Pandas, Matplotlib, Scikit.	
TASK– 2	**3H**
Perform Data exploration and pre-processing in Python.	
TASK– 3	**3H**
Perform Feature Engineering and Feature Selection Methods.	
TASK– 4	**3H**
Implement and demonstrate the FIND-S algorithm for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .CSV file.	
TASK– 5	**3H**
Implementation of Linear and Logistic Regression.	
TASK– 6	**3H**
Implementation of K means algorithm.	
TASK– 7	**3H**
For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples.	

TASK– 8	3H
Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.	
TASK– 9	**3H**
Build an Artificial Neural Network by implementing the Backpropagation algorithm and test the same using appropriate data sets.	
TASK– 10	**3H**
Write a program to implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets.	
TASK– 11	**3H**
Assuming a set of documents that need to be classified, use the naïve Bayesian Classifier model to perform this task. Built-in Java classes/API can be used to write the program. Calculate the accuracy, precision, and recall for your data set.	
TASK– 12	**3H**
Write a program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set. You can use Java/Python ML library classes/API.	
Total hours:	**36 hours**
Additional Experiments: Sentimental Analysis, Time series	
Virtual Labs: COLAB,JUPITOR NOTE BOOK	
Text Book(s): Ethem Alpaydın, Introduction to machine learning, second edition, MIT press.	
Reference Book(s): Android Programming by B.M Harwani, Pearson Education, 2013.	

NARAYANAENGINEERINGCOLLEGE::GUDUR								
SEM-VI	**WEB TECHNOLOGIES LABORATORY**							**R21**
Course Code	**Hours/Week**			**Total hrs**	**Credit**	**Max Marks**		
	L	**T**	**P**		**C**	**CIE**	**SEE**	**TOTAL**
21CS2509	0	0	3	36	1.5	40	60	100

Pre-requisite: Internet Programming ,Fundamental of Computing and Programming

Course Objectives:

- Demonstrate the role of languages like HTML, DHTML, CSS, XML, Javascript, ASP and protocols in the workings of the web and web applications.

Course Outcomes: After successful completion of the course, the student will be able to:

CO1	Develop web pages using HTML, DHTML and Cascading Styles sheets (BL-6)
CO2	Develop a dynamic web pages using JavaScript (client side programming).(BL-6)
CO3	Develop an interactive web applications using PHP. (BL-6)
CO4	Build and consume web services. (BL-3)

CO-PO Mapping

CO	**PO**												**PSO**	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	1		2										3	3
CO2	2		2										3	3
CO3	2		3		2								3	3
CO4	2		2		1								3	3

1-Low, 2-Medium, 3- High

COURSE CONTENT	**CO**
Task-1	
a. Create a web page to show books information in library using tables, lists and images. b.Create an html page with different types of frames such as floating frame, navigation frame & mixed frame.	CO 1
Task-2	
b. To create a simple student bio-data form. it should contain the following name (text box), address (multiline text box), gender (radio button male, female),skill sets known (check boxes – c, c++,java, C#etc), extracurricular activities (text box), nationality (combo box) ,submit and reset button.	CO 1
TASK-3	
Create a webpage to embeda map along with hotspot, frames & links.	CO 2
TASK-4	
Create a webpage using an embedded, external and inline CSS file.	CO 2

TASK-5	
Create an online job registration page and validate the form using java script	CO 2
TASK-6	
a. Write a JavaScript to define a user defined function for sorting the values in an array. b. Write a JavaScript code to handle different types of events.	CO 3
TASK-7	
Write a JavaScript code for creating dynamic tables and filling data through form fields.	CO 4
TASK-8	
a. Write a Php code to find the factorial of given number b. Write a Php code to find length of given string	CO 4
TASK –09	
a. Write a php code to show the access of Super globals. Write a php to design a form and validate the form fields	CO 4
TASK –10	
a. Create a login page using php and store the login details using mysql. Create an online application in any of the web application like PHP for Tourism management like the available trip details in season based.	CO 4
ADDITIONAL EXPERIMENTS	
TASK–11	CO 4
Case study on Student Result Entry and Viewing using HTML, CSS, PHP and MySQL	
TASK–12	
Case study on Core Banking Operations(CRUD) using HTML, CSS, PHP and MySQL	CO 4
Virtual Labs: https://ds1-iiith.vlabs.ac.in/List%20of%20experiments.html	

TEXT BOOK:

1. Harvey & Paul Deitel &Associates, Harvey Deitel and Abbey Deitel, "Internet and World Wide Web-How to Program", Fifth Edition, PearsonEducation, 2011.
2. The Joy of PHP Programming: A Beginner's Guide – by Alan Forbes

REFERENCES:

1. Gopalan N. P. and Akil and eswariJ., Web Technology, Prentice Hall of India, 2011.
2. Uttam K. Roy, WebTechnologies, Oxford University Press, 2011.
3. Head First jQuery, Ryan Benedetti and Ronan Cranley

SEMESTER –VII

Course Code	Category	Course Title	Contact Periods per week				Credits	Scheme of Examination Max. Marks		
			L	T	P	Total		Int. Marks	Ext. Marks	Total marks
21CS2013	PC	Cryptography and Network Security	3	0	0	3	3	40	60	100
21CS2014	PC	Mobile Application Development	3	0	0	3	3	40	60	100
21CS2015	PC	Deep Learning	2	0	0	2	2	40	60	100
	OE	Open Elective IV	3	0	0	3	3	40	60	100
21CS4016-20	PE	Professional Elective IV	3	0	0	3	3	40	60	100
21CS4021-25	PE	Professional Elective V	3	0	0	3	3	40	60	100
21CS2510	PC	Mobile Application Development Lab	0	0	2	2	1	40	60	100
21CS2511	PC	Deep Learning Lab	0	0	3	3	1.5	40	60	100
21CD6005	SC	Career Competency Development V	0	0	2	2	1	40	60	100
21CC6501	SC	Skill Development Training	0	0	2	2	1	40	60	100
20CS7502	PR	Internship II/On job Training/Comm. Service Project	0	0	0	0	1.5	40	60	100
		Counselling/Mentoring	0	0	1	1	0	--	--	--
		Sports/Hobby Clubs/Activities	0	0	2	2	0	--	--	--
		Activity Point Programme	During the Semester				20 Pts			
			17	**0**	**12**	**29**	**23**	**440**	**660**	**1100**

NARAYANAENGINEERINGCOLLEGE::GUDUR								
SEM-VII	**CRYPTOGRAPHY AND NETWORK SECURITY**							**R21**
Course Code	**Hours/Week**			**Total hrs**	**Credit**	**Max Marks**		
	L	**T**	**P**		**C**	**CIE**	**SEE**	**TOTAL**
21CS2013	3	0	0	50	3	40	60	100

COURSECONTENT		
MODULE– 1		**10H**
Attacks on Computers and Computer Security: Introduction, The need for security, Principles of security, Types of Security attacks, Security services, Security Mechanisms, A model for Network Security Cryptography, plain text and cipher text, substitution techniques, transposition techniques, encryption and decryption, symmetric and asymmetric key cryptography, Steganography.		
MODULE– 2		**9H**
Symmetric key Ciphers: Block Cipher principles &Algorithms (DES, AES, Blowfish), Block cipher modes of operation, Stream ciphers, Key distribution. **Asymmetric key Ciphers**: Principles of public key cryptosystems, Algorithms (RSA, Diffie Hellman, ECC), Key Distribution.		
MODULE– 3		**12H**
Message Authentication Algorithms and Hash Functions: Authentication requirements, Functions, Message authentication codes, Hash Functions, Secure hash algorithm, Whirlpool, HMAC, CMAC, Digital signatures, knapsack algorithm.		
MODULE– 4		**9H**
E-Mail Security: Pretty Good Privacy, S/MIME IP Security: IP Security overview, IP Security architecture, Authentication Header, encapsulating security payload, security associations, key-management.		
MODULE– 5		**10H**
Web Security: Web security considerations, Secure Socket Layer and Transport Layer Security, Secure electronic transaction Intruders, Virus and Firewalls: Intruders, Intrusion detection, password management, Virus and related threats, Firewall design principles, Types of firewalls Case Studies on Cryptography and security: Secure Inter-branch Payment Transactions, Virtual Elections		
	Total hours:	**50 hours**

TEXT BOOK(S):

1. William Stallings, "Cryptography and Network Security", 5th Edition, Pearson Education, 2011.
2. Bernard Menezes "Network Security and Cryptography", 1stEdition, CENGAGE Learning, 2010.

REFERENCES:

1. C K Shyamala, N Harini, Dr T R Padmanabhan, Wiley India, "Cryptography and Network Security", 1st Edition, Wiley India Pvt Ltd, 2011.
2. Forouzan Mukhopadhyay "Cryptography and Network Security", 2nd Edition, Mc Graw Hill, 2010.
3. Mark Stamp, Wiley India, "Information Security, Principles and Practice", 2nd Edition, Wiley, 2011

NARAYANAENGINEERINGCOLLEGE::GUDUR								
SEM-VII	**MOBILE APPLICATION DEVELOPMENT**							**R21**
Course Code	**Hours/Week**			**Total hrs**	**Credit**	**Max Marks**		
	L	**T**	**P**		**C**	**CIE**	**SEE**	**TOTAL**
21CS2014	3	0	0	48	3	40	60	100

COURSECONTENT		
MODULE– 1	**Introduction to Android**	**12H**
TheAndroid4.1jellyBeanSDK, Understanding the Android Software Stack, installing the Android SDK, Creating Android Virtual Devices, Creating the First Android Project, Using the Text view Control, Using the Android Emulator, Launching Android Applications on a Handset.		
MODULE-2	**Basic Widgets**	**10H**
The Role of Android Application Components, Utility of Android API, Overview of the Android Project Files, Understanding Activities, Role of the Android Manifest File, Creating the User Interface, Commonly Used Layouts and Controls, Event Handling, Displaying Messages Through Toast, Creating and Starting an Activity, Using the Edit Text Control, Choosing Options with Checkbox, Choosing Mutually Exclusive Items Using Radio Buttons.		
MODULE-3	**Building Blocks for Android Application Design**	**9H**
Introduction to Layouts, Linear Layout, Relative Layout, Absolute Layout, Using Image View, Frame Layout, Table Layout, Grid Layout, Adapting to Screen orientation. Utilizing Resources and Media Resources, Creating Values Resources, Using Drawable Resources, Switching States with Toggle Buttons, Creating an Images Switcher Application, Scrolling Through Scroll View, playing Audio, Playing Video		
MODULE-4	**Selection widgets And Fetching Information Using Dialogs and Fragments**	**9H**
Using List View, Using the Spinner control, Using the Grid View Control, Dialogs, Selecting the Date and Time in One Application, Fragments, Creating Special Fragments**.**		
MODULE-5	**Building Menus**	**8H**
Creating Interface Menus and Action Bars, Menus and Their Types, Creating Menus Through XML, Applying a Context Menu to a List View, Using the Action Bar, SqLite, Database Applications.		
	Total hours:	**48 hours**

TEXT BOOK(S):

1. B. MHarwani, Android Programming, Pearson Education.
2. Lauren Darcey and Shane Conder, "Android Wireless Application Development", 2nd edition, Pearson Education.

REFERENCES:

Professional Android Application Development, Wiley India Private Limited.

1. Dawn Griffiths, David Griffiths, "Head First Android Development: A Brain-Friendly Guide", Second Edition, O'ReillyMedia, 2017.
2. James C Sheusi, Android application Development for Java Programmers, Cengage
3. Learning
4. Reto Meier, Professional Android 4 applications development, Wiley India.
5. Wei-Meng Lee, BeginningAndroid4applicationsdevelopment, Wiley India.

NARAYANA ENGINEERING COLLEGE::GUDUR								
	DEEP LEARNING							R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
21CS4015	2	0	0	50	2	40	60	100

COURSE CONTENT		
MODULE – 1		**10H**
Linear Algebra: Scalars, Vectors, Matrices and Tensors, Matrix operations, types of matrices, Norms, Eigen decomposition, Singular Value Decomposition, Principal Components Analysis. Probability and Information Theory: Random Variables, Probability Distributions, Marginal Probability, Conditional Probability, Expectation, Variance and Covariance, Bayes' Rule, Information Theory. Numerical Computation: Overflow and Underflow, Gradient-Based Optimization, Constrained Optimization, Linear Least Squares.		
MODULE – 2		**10H**
Machine Learning: Basics and Under fitting, Hyper parameters and Validation Sets, Estimators, Bias and Variance, Maximum Likelihood, Bayesian Statistics, Supervised and Unsupervised Learning, Stochastic Gradient Descent, Challenges Motivating Deep Learning. Deep Feed forward Networks: Learning XOR, Gradient-Based Learning, Hidden Units, Architecture Design, Back-Propagation and other Differentiation Algorithms		
MODULE – 3		**10H**
Regularization for Deep Learning: Parameter Norm Penalties, Norm Penalties as Constrained Optimization, Regularization and Under-Constrained Problems, Dataset Augmentation, Noise Robustness, Semi-Supervised Learning, Multi-Task Learning, Early Stopping, Parameter Tying and Parameter Sharing, Sparse Representations, Bagging and Other Ensemble Methods, Dropout, Adversarial Training, Tangent Distance, Tangent Prop and Manifold Tangent Classifier. Optimization for Training Deep Models: Pure Optimization, Challenges in Neural Network Optimization, Basic Algorithms, Parameter Initialization Strategies, Algorithms with Adaptive Learning Rates, Approximate Second-Order Methods, Optimization Strategies and Meta-Algorithms		
MODULE – 4		**10H**
Convolutional Networks: The Convolution Operation, Pooling, Convolution, Basic Convolution Functions, Structured Outputs, Data Types, Efficient Convolution Algorithms, Random or Unsupervised Features, Basis for Convolutional Networks		
MODULE – 5		**10H**
Sequence Modelling: Recurrent and Recursive Nets: Unfolding Computational Graphs, Recurrent Neural Networks, Bidirectional RNNs, Encoder-Decoder Sequence-to-Sequence Architectures, Deep Recurrent Networks, Recursive Neural Networks, Echo State Networks, LSTM, Gated RNNs, Optimization for Long-Term Dependencies, Auto encoders, Deep Generative Models		
	Total hours:	**50 hours**

TEXTBOOK:

1. Ian Goodfellow, YoshuaBengio, Aaron Courville, “Deep Learning”, MIT Press, 2016.
2. Josh Patterson and Adam Gibson, “Deep learning: A practitioner's approach”, O'Reilly Media, First Edition,2017

REFERENCES:

1. Fundamentals of Deep Learning, Designing next-generation machine intelligence algorithms, Nikhil Buduma, O’Reilly, Shroff Publishers, 2019.
2. Deep learning Cook Book, Practical recipes to get started Quickly, O’Reilly, 2019

NARAYANAENGINEERINGCOLLEGE::GUDUR								
SEM-VII	**MOBILE APPLICATION DEVELOPMENT LABORATORY**							**R21**
Course Code	**Hours/Week**			**Total hrs**	**Credit**	**Max Marks**		
	L	**T**	**P**		**C**	**CIE**	**SEE**	**TOTAL**
21CS2510	0	0	2	38	1	40	60	100

List of Experiments		
TASK– 1	**Android installations**	**3H**
Setup the Development environment to develop Android Applications		
TASK– 2	**Hello World Application.**	**3H**
Create "Hello World" Application.		
TASK– 3	**Using the Activity class**	**1H**
Create an application using the Activity class.		
TASK– 4	**Edit Text control.**	**3H**
Create an application using Edit Text control.		
TASK– 5	**Check Box control.**	**3H**
CreatinganapplicationthatallowschoosingoptionsusingCheckBoxcontrol.		
TASK– 6	**Radio Button control**	**3H**
Creating an application that allows choosing options using Radio Button control		
TASK– 7	**Linear Layout**	**3H**
Create an application using Linear Layout		
TASK– 8	**Relative Layout**	**3H**
Create an application using Relative Layout		
TASK– 8	**Relative Layout**	**3H**
Create an application using Relative Layout		
TASK– 9	**Absolute Layout**	**3H**
Create an application using Absolute Layout		
TASK– 10	**Play Audio and Video clips**	**3H**
Create an application to play Audio and Video clips		
TASK– 11	**Using Spinner.**	**3H**
Create an application that allows choosing options using Spinner.		
TASK– 12	**Menus**	**3H**
Create an application using Menus.		
Additional Experiments: 3H		
TASK-13	**Radio Button control**	**1H**
Creating an application that allows choosing options using two sets of Radio Button controls.		
TASK-14	**Action Bar**	**1H**
1. Create an application using Action Bar. 2. Create an application to display a Drop-Down List Action Bar.		
	Total hours:	**38 hours**

TEXT BOOK:

1. Android Programming by B.M Harwani, Pearson Education, 2013.
2. Lauren Darcey and Shane Conder, "Android Wireless Application Development", Pearson Education, 2nd ed. (2011).
3. Professional Android Application Development, Wiley India Private Limited.

REFERENCES:

1. Dawn Griffiths, David Griffiths, "Head First Android Development: A Brain-Friendly Guide", Second Edition, O'Reilly Media, 2017.ISBN:978-1491974056.
2. Android application Development for Java Programmers, James C Sheusi, Cengage Learning
3. Android In Action by w. Frank Ableson, RobiSen, ChrisKing, C.EnriqueOrtiz., Dream tech.
4. Professional Android 4 applications development, RetoMeier, Wiley India, 2012.
5. Beginning Android 4 applications development, Wei-Meng Lee, Wiley India, 2013[2008], [6th Edition], Java How to Program, Pearson Ed.

NARAYANA ENGINEERING COLLEGE:: GUDUR								
SEM-VII	**DEEP LEARNING LABORATORY**							**R21**
Course Code	**Hours / Week**			**Total hrs**	**Credit**	**Max Marks**		
	L	**T**	**P**		**C**	**CIE**	**SEE**	**TOTAL**
21CS2511	0	0	3	38	1.5	40	60	100

List of Experiments	
TASK – 1	**3H**
Implementation of different activation functions to train Neural Network	
TASK – 2	**3H**
Build a deep neural network model start with linear regression using a single variable	
TASK – 3	**3H**
Write a program to convert speech into text	
TASK – 4	**3H**
Write a program to convert text into speech	
TASK – 5	**3H**
Write a program to convert video into frames	
TASK – 6	**3H**
Write a program for Time-Series Forecasting with the LSTM Model	
TASK – 7	**3H**
Build a feed forward neural network for prediction of logic gates	
TASK – 8	**3H**
Write a program to implement deep learning Techniques for image segmentation	
TASK – 9	**3H**
Write a program for object detection using image labeling tools	
TASK – 10	**3H**
Write a program to predict a caption for a sample image using LSTM	
TASK – 11	**3H**
Write a program for character recognition using CNN.	
TASK – 12	**3H**
Write a program to Implement Chatbot using bi-directional LSTMs.	
Additional Experiments	
TASK – 13	**1H**
Image Captioning with Vanilla RNNs	
TASK – 14	**1H**
Image Captioning with LSTMs	
Total hours:	**38 hours**

TEXT BOOKS:

1.Navin Kumar Manaswi ,Deep Learning with Applications Using Python Chatbots and Face, Object, and Speech Recognition With TensorFlow and Keras , Apress,2018.

REFERENCES:

1.Ian Goodfellow, Yoshua Bengio, Aaron Courville, “Deep Learning”, MIT Press,2016.

OPEN ELECTIVES(OE)

NARAYANA ENGINEERING COLLEGE:GUDUR								
	DATA STRUCTURES AND ALGORITHMS							R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
21CS3001	3	0	0	48	3	40	60	100

Pre-requisite: Knowledge of Mathematics, Computer Programming, Analytical & Logical Skills

Course Objectives:

1. To explain efficient storage mechanisms of data for an easy access.
2. To design and implementation of various basic and advanced data structures.
3. To introduce various techniques for representation of the data in the real world.
4. To develop applications using data structures.
5. To pertain knowledge on improving the efficiency of algorithm by using suitable data structure.

Course Outcomes: After successful completion of the course, the student will be able to:

CO 1	Understand basic concepts of data structures and algorithm analysis. (BL - 2)
CO 2	Develop the applications using stacks and queues. (BL - 3)
CO 3	Demonstrate the use of linked lists. (BL - 2)
CO 4	Apply tree, graph data structures for various applications. (BL - 3)
CO 5	Implement algorithms for sorting, searching, and hashing methods. (BL - 3)

CO-PO Mapping

CO	PO												PSO	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO 1	1	1	2										1	
CO 2	2	3	2	2									2	1
CO 3	2	2	3	2	2								3	2
CO 4	2	2	2	1	1							2	3	2
CO 5	2	1	2	1								1	2	2

1: Low, 2-Medium, 3- High

COURSE CONTENT		
MODULE – 1	**Introduction to Data Structures**	**10H**
Introduction: Overview of Data Structures, Implementation of Data Structures, Algorithm Specifications, Analysis of an Algorithm, Asymptotic Notations, Time-Space trade off. **Arrays:** One-Dimensional, Multi-Dimensional, Pointer Arrays.		
At the end of the Module 1, students will be able to: 5. Understand the linear and non-linear data structures. (BL - 2) 6. Understand the time and space complexities of an algorithm. (BL - 2) 7. Illustrate representation of data using Arrays. (BL - 2)		
MODULE -2	**Stacks and Queues**	**9H**
Stacks: Introduction, Representation of a Stack, Stack Operations, Applications of Stacks. **Queues:** Introduction, Representation of a Queue, Queue Operations, Circular Queue, Applications of Queues.		
At the end of the Module 2, students will be able to: 7. Explain stack ADT and its operations. (BL - 2) 8. Understand the expression evaluation using stacks. (BL - 2) 9. Implement various queue structures. (BL - 3)		
MODULE-3	**Linked Lists**	**9H**
Introduction, Singly linked lists, Doubly Linked Lists, Circular Linked Lists, Linked Stacks and Queues, Applications of Linked Lists.		
At the end of the Module 3, students will be able to: 4. Understand basics concepts of linked lists. (BL - 2) 5. Illustrate various structures of linked lists. (BL - 2) 6. Understand the concept of dynamic memory management. (BL - 2)		
MODULE-4	**Trees & Graphs**	**10H**
Trees-Introduction, Basic Terminologies, Definition and concepts, Representation of Binary Tree, operations on a Binary Trees, Binary Search Trees, Height Balanced Binary Tree.Graph Terminologies, Representation of Graphs, Graph Operations, Shortest Paths – Warshall's, Floyd's and Dijkstra's algorithms, Topological Sorting.		
At the end of the Module 4, students will be able to: 4. Understand the concept of trees. (BL - 2) 5. Compare different tree structures. (BL - 2) 6. Explain the importance of Graphs for solving problems. (BL - 2) 7. Understand graph traversal methods. (BL - 2) 8. Implement algorithms to identify shortest path. (BL - 3)		
MODULE-5	**Sorting, Searching and Hash Tables**	**10H**

Sorting: Introduction, Bubble Sort, Selection Sort, Quick Sort. Searching: Introduction, Basic Terminology, Linear Search and Binary Search Techniques. Hash Table: Hashing Techniques, Collision Resolution Techniques, Closed Hashing, Open Hashing.
At the end of the Module 6, students will be able to: 1. Implement the sorting algorithms (BL - 3) 2. Select the appropriate sorting algorithm for a given application (BL - 3) 3. Understand the concept of Hash Table (BL - 2) 4. Explain searching techniques. (BL - 2)

Total hours:	**48 hours**

Content beyond syllabus: Heap Sort, Insertion Sort, Merge Sort, Optimum Sorting Algorithms

Text Book(s):

1. Samanta, "Classic Data Structures", 2nd Edition, Prentice-Hall of India, Pvt. Ltd., India, 2012.
2. Ellis Horowitz and Sartaj Sahni, "Fundamentals of Data Structures in C", 2nd Edition, Universities Press , 2008.

Reference Books:

1. Narasimha Karumanchi, Data Structures and Algorithms Made Easy, Careermonk Publications, 2016
2. Peter Bras, "Advanced Data Structures", Cambridge University Press, 2014.
3. RS Salaria, Data Structures, 3rd Edition, Khanna Publishing House, 2017.
8. YashwantKanetkar, Data Structures through C,3rd Edition, BPB Publications, 2019.
9. RB Patel, Expert Data Structures with C, Khanna Publications, 2019.
10. Richard F. Gilberg, Behrouz A. Forouzan, Data Structures A Pseudo code Approach with C, Second Edition, Cengage Learning.
11. Ananda Rao Akepogu, Radhika Raju Palagiri,Data Structures and Alg. Using C++ ,

NARAYANA ENGINEERING COLLEGE:GUDUR								
	INTRODUCTION TO PYTHON							R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
21CS3002	3	0	0	48	2	40	60	100

Pre-requisite: Knowledge of Mathematics and Basic Programming Language

Course Objectives:

1. To learn the fundamentals of python.
2. To implement python programs for conditional loops and functions.
3. To handle the compound data using python lists, tuples, sets, dictionaries.
4. To learn the files, modules, packages concepts.
5. To introduce the concepts of class and exception handling using python.

Course Outcomes: After successful completion of the course, Student will be able to:

CO 1	Summarize the fundamental concepts of python programming. (BL - 2)
CO 2	Apply the basic elements and constructs the python to solve logical problems.(BL-3)
CO 3	Organize data using different data structures of python. (BL - 3)
CO 4	Implement the files modules and packages in programming. (BL - 3)
CO 5	Apply object-oriented concepts to build simple applications. (BL - 3)

CO-PO Mapping

CO	PO												PSO	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	3	2	1	1								1		
CO2	1	3	2	2	1	2			1	1				
CO3	1	1	3	2	2									
CO4	1	3	2	2										
CO5	1	3	2	2										

1: Low, 2-Medium, 3- High

COURSE CONTENT		
MODULE – 1	**Introduction to Python**	**10 H**
History of Python, Features of Python Programming, Applications of Python Programming, Running Python Scripts, Comments, Typed Language, Identifiers, Variables, Keywords, Input/output, Indentation, Data types, Type Checking, range( ), format( ), Math module.		
At the end of the Module 1, students will be able to: 1. Learn the basics of python. (BL - 1) 2. Write the python programs. (BL - 1) 3. Understand concept of type checking. (BL - 2)		
MODULE -2	**Operators Expressions and Functions**	**10 H**
Arithmetic, Assignment, Relational, Logical, Boolean, Bitwise, Membership, Identity, Expressions and Order of Evaluations, Control Statements. Defining Functions, Calling Functions, Anonymous Function, Fruitful Functions and Void Functions, Parameters and Arguments, Passing Arguments, Types of Arguments, Scope of variables, Recursive Functions.		
At the end of the Module 2, students will be able to: 1. Solve the problems using operators, conditional and looping. (BL - 3) 2. Solve the problems using the functions. (BL -3) 3. Apply the principle of recursion to solve the problems. (BL-3)		
MODULE-3	**Strings, Lists, Tuples, and Dictionaries**	**9 H**
Strings- Operations, Slicing, Methods, List- Operations, slicing, Methods, Tuple- Operations, Methods, Dictionaries- Operations, Methods, Mutable Vs Immutable, Arrays Vs Lists, Map, Reduce, Filter, Comprehensions.		
At the end of the Module 3, students will be able to: 1. Write programs for manipulating the strings. (BL - 1) 2. Understand the knowledge of data structures like Tuples, Lists, and Dictionaries.(BL - 2) 3. Select appropriate data structure of Python for solving a problem.(BL -3)		
MODULE-4	**Files, Modules and Packages**	**10 H**
Files- Persistent, Text Files, Reading and Writing Files, Format Operator, Filename and Paths, Command Line Arguments, File methods, Modules- Creating Modules, Import Statement, Form. Import Statement, name spacing, Packages- Introduction to PIP, Installing Packages via PIP(Numpy).		
At the end of the Module 4, students will be able to: 1. Understand the concepts of files. (BL - 2) 2. Implement the modules and packages. (BL - 3) 3. Organize data in the form of files. (BL - 3)		
MODULE-5	**Object Oriented Programming, Errors and Exceptions**	**9 H**
Object Oriented Features, Classes, self variable, Methods, Constructors, Destructors,		

Inheritance, Overriding Methods, Data hiding, Polymorphism. Difference between an error and Exception, Handling Exception, try except block, Raising Exceptions.	
At the end of the Module 5, students will be able to: 1. Apply object orientation concepts.(BL -3) 2. Apply the exception handling concepts. (BL -3) 3. Implement OOPs using Python for solving real-world problems. (BL -3)	
Total hours:	**48 Hours**

Content Beyond Syllabus: Turtle Module, GUI Programming, Matplotlib, Databases.

Text Book(s):

1. Vamsi Kurama, Python Programming: A Modern Approach, Pearson, 2017.
2. Allen Downey, Think Python, 2ndEdition,Green Tea Press

Reference Books :

1. R. Nageswara Rao, "Core Python Programming", 2nd edition, Dreamtech Press, 2019.
2. Allen B. Downey, "Think Python", 2ndEdition, SPD/O'Reilly, 2016.
3. Martin C.Brown, "The Complete Reference: Python", McGraw-Hill, 2018.
4. Mark Lutz, Learning Python, 5th Edition, Orielly, 2013.
5. Wesley J Chun, Core Python Programming, 2nd Edition, Pearson, 2007
6. Kenneth A. Lambert, Fundamentals of Python, 1st Edition, Cengage Learning, 2015

NARAYANA ENGINEERING COLLEGE::GUDUR								
	OBJECT ORIENTED PROGRAMMING THROUGH JAVA							R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
21CS3003	3	0	0	48	3	40	60	100

Pre-requisite: Basic knowledge of programming.

Course Objectives:

6. To acquire knowledge on preliminaries of Java.
7. To provide sufficient knowledge on developing real world projects.
8. To demonstrate the principles of packages, inheritance, and interfaces.
9. To understand exception handling, Event handling and Multithreading.
10. To design and build Graphical User Interface applications.

Course Outcomes: After successful completion of the course, Student will be able to:

CO1	Understand Object Oriented Programming concepts. (BL-2)
CO2	Demonstrate the concepts of Arrays and Strings. (BL-2)
CO3	Construct programs on classes, inheritance, and polymorphism. (BL-3)
CO4	Develop packages and interfaces. (BL-3)
CO5	Apply multi-threading and graphical user interface concepts for real time applications. (BL-3)

CO-PO Mapping

CO	**PO**												**PSO**	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	3	2	2									1	3	2
CO2	2	3	2		1							1	1	2
CO3	2	2	3	2	1				1			2	1	2
CO4	2	2	2	3	2	1			1			2	1	1
CO5	2	2	2	3	2	1			1			2	2	3

1: Low, 2-Medium, 3- High

COURSE CONTENT		
MODULE – 1	**Basic concepts of java**	**9h**
The History and Evolution of java: OOP Concepts, History of java, The java Buzz words, The Evolution of java, Lexical issues. Data types, variables: Data types, Variables, The Scope and Life time of variables, Operators, Expressions, Control statements, Type conversion and casting, Command Line Arguments.		
At the end of the Module 1, students will be able to: 4. Describe the Purpose of Object oriented Programming Concepts.(BL-2) 5. Understand the importance of java. (BL-2) 6. Identify various basic components of java. (BL-2) 7. Implement programs on fundamental concepts of java. (BL-2)		
MODULE -2	**Arrays and String Handling**	**9h**
Arrays: Declaration, Initialization and accessing values, One-Dimensional Arrays, Multi-dimensional arrays, Alternative Array Declaration Syntax, var-arg methods. Strings: Explore String class, StringBuffer and StringBuilder classes.		
At the end of the Module 2, students will be able to: 4. Understand Arrays and accessing array values. (BL-2) 5. Demonstrate1-D and Multi-dimensional arrays. (BL-2) 6. Illustrate the String and StringBuffer Classes. (BL-2)		
MODULE-3	**Classes, Inheritance and polymorphism**	**10h**
Class fundamentals. Declaration objects, Assigning object reference variables, Introducing Methods, Constructors, "this" keyword, Garbage collection. Inheritance basics, Using Super keyword, Types of inheritance, Benefits, Member access rules, Constructor and calling sequence, Using abstract Classes, Using final keyword. Method overriding and overloading.		
At the end of the Module 3, students will be able to: 5. Understand the basic syntax for class fundamentals. (BL-2) 6. Demonstrate Access modifiers in Inheritance. (BL-2) 7. Compare "Method overloading and Method overriding". (BL-3)		
MODULE-4	**Packages and Exception Handling**	**9h**
Defining an interface, Implementing interface, Accessing interface properties. Defining Package, finding packages and class path, accessing Protection**.** Exception handling Fundamentals, exception types, Built-in Exceptions, Using try-catch-finally throw- throws keywords, creating your own Exception subclasses.		
At the end of the Module 4, students will be able to: 5. Demonstrate interface and its implementation (BL-2)		

6. Develop user defined packages. (BL-3)
7. Implement Exception Handling. (BL-3)

MODULE-5	Multi-Threaded Programming and I/O	11h

The java thread model, Thread Life Cycle, The main thread, creating a Thread, Creating Multiple Threads, Using isalive() and join().MVC architecture, creating a window, components and containers, Basics of components, points and rectangles, visual characteristics of components, Defining color, creating cursors, selecting Font, swing components , Layout Managers.

At the end of the Module 4, students will be able to:

1. Demonstrate Multi-Threaded Programming. (BL-2)
2. Understand MVC architecture. (BL-2)
3. Illustrate components of GUI in java. (BL-2)

Total hours: 48 h

Content beyond syllabus:

1. Client /Server Communication applications (Servlets, jsp).
2. Database connectivity (JDBC).

Self-Study:

Contents to promote self-Learning:

Text Book(s):

1. Herbert Schildt, "Java The complete reference", 9th edition, McGraw Hill Education (India) Pvt. Ltd.
2. Ivor Horton, Beginning Java 2, JDK 5th Edition, Wiley dreamtech.

Reference Book(s):

1. An introduction to java programming and object oriented application development, R AJohson-Thomson.
2. Introduction to java programming 6thEdition, Y Daniel liang, Pearson Education.
3. Java programming: A practical approach, C.Xavier, TMH, First edition,2011.
4. Thinking in Java ,Bruce Eckel, 2nd Edition, Pearson Education
5. Java How to Program, H.M Dietel and P.J Dietel,6th Edition, Pearson Ed.
6. Introduction to Java programming-comprehensive, Y. Daniel Liang, Tenth Edition,Pearson ltd 2015.
7. E Balagurusamy, Programming With Ja9v2a : A Primer 5th Edition Tata McGraw Hill.

NARAYANA ENGINEERING COLLEGE:GUDUR								
21CS3004	ADVANCED JAVA PROGRAMMING							R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
21CS3004	3	0	0	48	3	40	60	100

Pre-requisite: Knowledge of core concepts of java programming.

Course Objectives:

1. To provide knowledge on console, GUI and Web based applications.
2. To understand the java technologies for multi-tier enterprise application development.
3. To practice applications development on Integrated Development Environment.
4. To perform operations on database using java database connectivity.
5. To examine the working principles of real time enterprise applications.

Course Outcomes: After successful completion of the course, Student will be able to:

CO1	Implement simple Web Applications and networking API.(BL 2)
CO2	Develop database applications using JDBC.(BL 3)
CO3	Understand the dynamic request and response model using Servlets .(BL 2)
CO4	Design enterprise application using Java Server Pages(JSP).(BL 3)
CO5	Implement Web applications using struts and Spring(BL 3)

CO-PO Mapping

CO	PO												PSO	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	3	2	2	2									2	2
CO2	2	2	2	1	2								2	1
CO3	1	2	2	2	1	1						2	2	1
CO4	2	1	2	1								2	1	1
CO5	2	2	1	2	2							2	2	2

1: Low, 2-Medium 3- High

COURSE CONTENT		
MODULE – 1	**Introduction to J2EE and Networking**	**10h**
Java Enterprise Edition: Java Platform, J2EE Architecture Types, Explore Java EE Containers, Types of Servers in J2EE Application, HTTP Protocols and API, Request Processing in Web Application, Web Application Structure, Web Containers and Web Architecture Models. **Java Networking:** Network Basics and Socket overview, TCP/IP client sockets, URL,TCP/IP server sockets, Datagrams, java.net package Socket, ServerSocket, InetAddress, URL, URLConnection**.**		
At the end of the Module 1, students will be able to: 5. Understand J2EE Architecture Types, containers and servers. (BL 2) 6. Gain knowledge on HTTP Protocols and APIs. (BL 2) 7. Discuss web applications and models. (BL 2) 8. Explain TCP/IP client server sockets programming. (BL 2)		
MODULE -2	**JDBC Programming**	**9h**
The JDBC Connectivity Model, Database Programming :Connecting to the Database, Creating a SQL Query, Getting the Results, Updating Database Data, Error Checking and the SQL Exception Class, The SQL Warning Class, The Statement Interface, Prepared Statement, Callable Statement The Result Set Interface, Updatable Result Sets, JDBC Types, Executing SQL Queries, Result Set Meta Data, Executing SQL Updates, Transaction Management.		
At the end of the Module 2, students will be able to: 1. Prepare The JDBC Connectivity Model. (BL 3) 2. Practice on PreparedStatement, Callable Statement and ResultSet Interface. (BL 3) 3. Explain JDBC Types. (BL 2) 4. Implement SQL Queries &Transaction Management. (BL 2)		
MODULE-3	**Servlet API and Overview**	**10h**
Overview of Servlet, Servlet Life Cycle, HTTP Methods Structure and Deployment descriptor Servlet Context and Servlet Config interface, Attributes in Servelt Request Dispache rinterface, The Filter API: Filter, Filter Chain. Using the Generic Servlet Class**.** Understanding state and session, Understanding Session Timeout and Session Tracking, URL Rewriting.		
At the end of the Module 3, students will be able to: 1. Understand Servlet Life Cycle. (BL 2) 2. Differentiate ServletContext and ServletConfig interface. (BL 2) 3. Understand Config Cookies and Session Management. (BL 2) 4. Differentiate the GenericServlet and HTTP Servlet Class. (BL 2)		
MODULE-5	**Java Server Pages**	**9h**

The Problem with Servlets, Life Cycle of JSP Page, JSP Processing, JSP Application Design with MVC, Setting Up the JSP Environment JSP Directives, JSP Action, JSP Implicit Objects JSP Form Processing, JSP Session and Cookies Handling.JSP Session Tracking JSP Database Access, JSP Standard Tag Libraries, JSP Custom Tag, JSP Expression Language, JSP Exception Handling, JSP XML Processing.

At the end of the Module 4, students will be able to:

1. Understand Life Cycle of JSP Page. (BL 2)
2. Explain MVC architecture and JSP Environment. (BL 2)
3. Construct JSP with DATABASES and exception handling. (BL 3)
4. Understand the role of XML in JSP. (BL 2)

MODULE-5	**Struts and Spring Frame Work**	**10h**

Basics & Architecture – Request Handling Life Cycle - Building a simple struts– Configuration, Actions, Interceptors, Results, Struts2 Tag Libraries, Struts2 XML Based Validations - Database Access. Overview of Spring, Spring Architecture, bean life cycle, XML Configuration on Spring, Aspect – oriented Spring, Managing Database, Managing Transaction.

At the end of the Module 5, students will be able to:

1. Explain struts frame work. (BL 2)
2. Implement the Struts Framework. (BL 3)
3. Understand Spring Architecture(BL-2)
4. Implementation of spring to build web applications(BL-3).

Content beyond syllabus: java mobile application development.

Text Book(s):

1. Black Book " Java server programming" J2EE, 1st ed., Dream Tech Publishers, 2008.
2. James Keogh, Complete Reference J2EE,mcgraw publication

Reference Book(s):

1. Matthew Scarpino, Hanumant Deshmukh, JigneshMalavie SCWCD, , Manning publication
2. Cay Horstmann and Gary Cornell, Core Java, Volume II: Advanced Features, Pearson Publication
3. Christian Bauer, Gavin King, Java Persistence with Hibernate,
4. Craig walls, Spring in Action, 3rdedition , Manning Publication
5. Jeff Linwood and Dave Minter Hibernate 2nd edition, Beginning Après publication
6. Kito D. Mann, Java Server Faces in Action, Manning Publication
7. Maydene Fisher, Jon Ellis, Jonathan Bruce, JDBC™ API Tutorial and Reference, Third Edition, Addison Wesley.
8. Giulio Zambon, Beginning JSP, JSF and9T5omcat, Apress.
9. Anghel Leonard, JSF2.0 CookBook, PACKT publication

NARAYANA ENGINEERING COLLEGE::GUDUR								
	DATABASES MANAGEMENT SYSTEM							R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
21CS3005	3	0	0	48	3	40	60	100

Pre-requisite: Knowledge of computer programming.

Course Objectives:

1. To Understand the basic concepts and the applications of database systems.
2. To design databases using data modeling and Logical database design techniques.
3. To construct database queries using relational algebra and calculus and SQL.
4. To explore implementation issues in database transaction.
5. To familiarize issues of concurrency control and transaction management.

Course Outcomes: On successful completion of the course, student will be able to:

CO 1	Demonstrate the fundamental elements and the applications of database management system (BL-2)
CO 2	Analyze the integrity constraints for relevant problems in database system. (BL-4)
CO 3	Construct the SQL queries to create, manipulate and extract the information in database system (BL - 3)
CO 4	Illustrate the concept of Normalization to produce a good database design in database design process. (BL - 2)
CO 5	Demonstrate Transactions and concurrency control in maintaining the database's integrity in database Systems. (BL - 2)

CO-PO Mapping

CO	PO												PSO	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	3	1											1	
CO2	3	2											1	
CO3	2	3											2	2
CO4	3	3	2										2	
CO5	3	2											2	

1: Low, 2-Medium, 3- High

COURSE CONTENT		
MODULE - 1	**Introduction to Database concepts and Modeling**	**10 H**
Introduction to Data bases, Purpose of Database Systems, View of Data, Data Models, Database Languages, Database Users, Database Systems architecture.Overview of Database Design, Beyond ER Design, Entities, Attributes and Entity sets, Relationships and Relationship sets, Conceptual Design with the ER Model.		
At the end of the Module 1, students will be able to: 4. Understand the Purpose of Database Systems, Data Models, View of Data. (BL-2) 5. Summarize the concept of Database Languages, Users, Architecture. (BL-2) 6. Design ER diagrams for given database. (BL-2)		
MODULE - 2	**Relational Model, Relational Algebra**	**9 H**
Introduction to the Relational Model – Integrity Constraints over Relations, Enforcing Integrity constraints, querying relational data, Logical data base Design, Views. Introduction to Relational algebra, selection and projection, set operations, renaming, joins, division.		
At the end of the Module 2, students will be able to: 4. Understand Basics of Relational Model. (BL-2) 5. Describe phases of Logical Database Design.(BL-2) 6. Explain the relational algebra operations on relations. (BL-2)		
MODULE - 3	**SQL**	**10 H**
SQL: Basic form of SQL Query, DDL, DML, Views in SQL, Joins, Nested & Correlated queries, Operators, Aggregate Functions, integrity Constraints.		
At the end of the Module 3, students will be able to: 4. Construct SQL queries in RDBMS. (BL-3) 5. Understand integrity and security Constraints in SQL (BL-2) 6. Construct PL/SQL programs in RDBMS. (BL-3)		
MODULE - 4	**Normalization**	**10 H**
Relational database design: Pitfalls of RDBD, Lossless join decomposition, Functional dependencies, Normalization for relational databases 1st, 2nd and 3rd normal forms.		
At the end of the Module 4, students will be able to: 1. Analyze functional dependencies. (BL-4) 2. Apply normal forms on functional dependencies. (BL-3) 3. Understand Multi Valued Dependencies and Join Dependencies (BL-2)		
MODULE - 5	**Transaction Management**	**9 H**
Transaction processing, Transaction Concept, Transaction State, Implementation of Atomicity and Durability, Concurrent Executions, Failure Classification, Recovery and Atomicity.Introduction to Index data structures, Hash-Based, Tree Based Indexing		
At the end of the Module 5, students will be able to: 1. Understand Atomicity and Durability, Concurrent Executions. (BL-2) 2. Discuss the concept of Transaction, Transaction State. (BL-2) 3. Discuss the Concurrency Control and various Protocols. (BL-2)		
	Total hours:	**48 Hours**
Content beyond syllabus: Embedded SQL Client/Server Database environment		

Web Database environment		
Self-Study: Contents to promote self-Learning:		
S.No	**Module**	**Reference**
1	Introduction to Database concepts and Modelling	https://nptel.ac.in/courses/106/105/106105175/ Week 1 – Lecture 1,2 https://nptel.ac.in/courses/106/105/106105175/ Week 1 – Lecture 3,4
2	Relational Model, Relational Algebra	https://nptel.ac.in/courses/106/106/106106220/ Week-3
3	SQL	https://nptel.ac.in/courses/106/105/106105175/ Week 3 – Lecture 6,7,8,9,10
4	Normalization	https://nptel.ac.in/courses/106/105/106105175/ Week 4 – Lecture 31,32,33,34,35
5	Transaction Management	https://nptel.ac.in/courses/106/105/106105175/ Week 7 – Lecture 6,7,8,9,10

Text Book(s):

1. Abraham Silberschatz, Henry F. Korth, S. Sudarshan, Database System Concepts, 6th Edition, Tata McGraw-Hill Publishing Company,2017.
2. Raghu Ramakrishnan, Database Management System, 3rd Edition, Tata McGraw-Hill Publishing Company, 2014.

Reference Book(s):

1. Peter Rob, A.Ananda Rao, Corlos Coronel, Database Management Systems (for JNTU), Cengage Learning, 2011.
2. Hector Garcia Molina, Jeffrey D. Ullman, Jennifer Widom, Database System Implementation, 1st Edition, Pearson Education, United States, 2000.
3. E. Ramez and Navathe, Fundamental of Database Systems, 7th Edition, Pearson Education
4. R.P. Mahapatra & Govind Verma, Database Management Systems, Khanna Publishing House, 2016.
5. 5Carlos Coronel and Steven Morris, Database Systems: Design, Implementation, and Management, 12th edition, Cengage Learning,2016.
6. John V. ,Absolute beginner's guide to databases, Petersen, QUE

Online Resources/ Web References:

https://www.coursera.org/learn/database-management
https://www.coursera.org/learn/sql-data-science
https://www.w3schools.com/sql/
https://www.youtube.com/watch?v=fHAfc7Hjq28&list=PLWPirh4EWFpGrpcMfZ6UcdI786QdtSxV8
https://www.youtube.com/watch?v=HwmEcudlv44&list=PL4OCRJojkV1jN-Ed6RkQpWfBvqe0utRd6
http://www.w3schools.in/dbms/

NARAYANA ENGINEERING COLLEGE:GUDUR								
	OPERATING SYSTEMS							R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
21CS3006	3	0	0	48	3	40	60	100

Pre-requisite: Fundamentals of computers

Course Objectives:

1. To understand the fundamental principles of the operating system, its services and Functionalities.
2. To illustrate the concepts of inter-process communication, synchronization and scheduling.
3. To understand different types of memory management viz. virtual memory, paging and segmentation.
4. To identify the reasons for deadlock and understand the techniques for deadlock detection, prevention and recovery.
5. To understand the need of Mass storage and protection mechanisms in computer systems.

Course Outcomes: After successful completion of the course, Student will be able to:

CO 1	Describe the concept operating system and operating system design. (BL-2)
CO 2	Analyze Process and CPU Scheduling, Process Coordination with concurrencies. (BL-3)
CO 3	Identify and evaluate Memory Management and Virtual Memory. (BL-3)
CO 4	Organize File System Interface. (BL-3)
CO 5	Understand Mass Storage Structure and Protection Mechanism. (BL-2)

CO-PO Mapping

CO	PO												PSO	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	1	2	2	2									2	
CO2		2	2	1										
CO3	3	1	2	1	1								1	
CO4	1	2	1		1									
CO5	3	2	1		2								2	

1: Low, 2-Medium, 3- High

COURSE CONTENT		
MODULE – 1	**Introduction**	**9H**
Computer system architecture, operating systems structure, operating systems operations; Evolution of operating systems: Simple Batch, multi programmed, time shared, parallel distributed systems, real time systems, special purpose systems, operating system services, user operating systems interface. Types of systems calls, system programs, protection and security, operating system design and implementation, operating systems structure**.**		
At the end of the Module 1, students will be able to: 1. Illustrate the structure of operating system and basic architectural components involved in operating system design. (BL-2) 2. Demonstrate how the computing resources are managed by the operating system. (BL-2) 3. Explain the objectives and functions of operating systems. (BL-2)		
MODULE -2	**Process and CPU scheduling, process coordination**	**10H**
The process, process state, process control block, threads; Process scheduling: Scheduling queues, context switch, preemptive scheduling, dispatcher, scheduling criteria, scheduling algorithms. Process synchronization, the critical section problem, synchronization hardware, semaphores and classic problems of synchronization, monitor. Deadlock characterization, methods of handling deadlocks, deadlock prevention, dead lock avoidance, dead lock detection and recovery from deadlock**.**		
At the end of the Module 2, students will be able to: 1. Contrast the process and a thread. (BL-2) 2. Develop applications to run in parallel either using process or thread models of different operating system. (BL-3) 3. Illustrate the various resource management techniques for timesharing and distributed systems. (BL-2) 4. Describe deadlock and deadlock mechanisms.(BL-2)		
MODULE-3	**Memory management and virtual memory**	**10H**
Swapping, contiguous memory allocation, paging, structure of page table. Segmentation with paging, virtual memory, demand paging; Performance of demand paging: Page replacement, page replacement algorithms, allocation of frames, thrashing**.**		
At the end of the Module 3, students will be able to: 1. Demonstrate the virtual memory, entities and attributes. (BL-3) 2. Illustrate the mapping from virtual memory address to physical address and vice-versa. (BL-3) 3. Identify how a shared memory area can be implemented using virtual memory addresses in different processes. (BL-3) 4. Contrast between Paging and Segmentation. (BL-2)		

MODULE-4	File system interface	9H
The concept of a file, access methods, directory structure, file system mounting, file sharing, protection, file system structure. File system structure, File system implementation, directory implementation, allocation methods, free space management**.**		
At the end of the Module 4, students will be able to: 1. List the mechanisms adopted for file distribution in applications. (BL-1) 2. Explain the need of memory management in operating systems and understand the limits of fixed memory allocation schemes. (BL-2) 3. Organize file management when designing or developing a new operating system. (BL-3)		
MODULE-5	**Mass-storage structure**	**10H**
Overview of mass storage structure, Disk structure, Disk attachment, Disk scheduling, Disk management, Swap space management, RAID structure, Stable storage implementation**.** goals of protection, principles of protection, domain of protection, access matrix, implementation of access matrix		
At the end of the Module 5, students will be able to: 1. Illustrate the fragmentation in dynamic memory allocation, and identify dynamic allocation approaches.(BL-2) 2. Illustrate how program memory addresses relate to physical memory addresses, memory management in base-limit machines, and swapping.(BL-2) 3. Compare RAID levels of memory.(BL-2) 4. Illustrate various disk scheduling algorithms.(BL-2) 5. Understand the access control and protection mechanisms. (BL-2)		
	Total hours:	**48 hours**

Content beyond syllabus:

Linux operating systems, Multiprocessor management systems, Unix features, real time operating systems, modern operating systems.

Text Book(s):

1. Abraham Silberschatz, Peter B. Galvin, Greg Gagne, "Operating System Principles",10th Edition, Wiley Student Edition, 2018.
2. William Stallings, "Operating System- Internals and Design Principles", 6th Edition, Pearson Education, 2002.

Reference Book(s):

1. D. M. Dhamdhere, "Operating Systems a Concept based Approach", 2nd Edition, Tata McGraw-Hill, 2006.
2. P.C.P. Bhatt, "An Introduction to Operating Systems", PHI Publishers.
3. G. Nutt, N. Chaki and S. Neogy, "Operating Systems", Third Edition, Pearson Education.
4. Andrew S Tanenbaum, "Modern Operating Systems", 3rd Edition, PHI, 2007.

NARAYANA ENGINEERING COLLEGE::GUDUR								
	COMPUTER NETWORKS							R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
21CS3007	3	0	0	48	3	40	60	100

Pre-requisite: Knowledge of Information Technology, Computer Organization & Architecture

Course Objectives:

6. To impart the core principles of Information Communication Technology.
7. To deliver background information on the key transmission technologies used in computer networks.
8. To convey dimensions of Network layer through Internet Protocol.
9. To provide an insight into the most widely used Transport Layer protocols
10. To teach the principles of Application Layer and its protocols.

Course Outcomes: On successful completion of the course, student will be able to:

CO 2	Choose suitable transmission media depending on requirements. (BL-2)
CO 3	Determine the errors in data transfer between source and destination. (BL-3)
CO 4	Obtain the skills of subnetting and routing mechanisms. (BL-2)
CO 5	Illustrate reliable, unreliable communication on public networks. (BL-3)
CO 6	Demonstrate elements of socket programming, principles of protocols.(BL-3)

CO-PO Mapping

CO	PO												PSO	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	2	2										1		
CO2	2	2	3	3									3	3
CO3	2	3	2										1	2
CO4	2	1											1	
CO5	2	1	1									1		1

1: Low, 2-Medium, 3- High

COURSE CONTENT		
MODULE – 1	**Physical Layer**	**(10H)**
Introduction: Data Communications, Networks, Network Types, Internet History, Standards and Administration, Protocol Layering, TCP/IP Protocol Suite, The OSI Model**,** Data and Signals, Digital Signals, Transmission Impairment, Data Rate Limits, Performance. Transmission Media: Introduction, Guided Media, Unguided Media.		
At the end of the Module 1, students will be able to: 1. Understand the basics of computer networks. (BL-2) 2. Summarize the concept of Internet and its standards. (BL-2) 3. Describe the picture of data communication with layered architecture. (BL-2) 4. Classify the elements of physical media used for data transmission. (BL-2)		
MODULE – 2	**Data-Link Layer & MAC**	**(9H)**
Introduction, Link-Layer Addressing, Error Detection and Correction: Checksum, CRC, Data Link Control (DLC)**:**DLC Services, Data-Link Layer Protocols, HDLC, PPP. Media Access Control (MAC): Random Access.		
At the end of the Module 2, students will be able to: 1. Explain link layer services. (BL-2) 2. Discuss Error Detection and Correction mechanisms. (BL-2) 3. Describe Data Link Control services and protocols. (BL-2) 4. Illustrate Media Access Control Protocols. (BL-3)		
MODULE – 3	**Network Layer**	**(10H)**
Network Layer: Network Layer Design Issues, Routing Algorithms: The Optimality Principle, Shortest Path Algorithm, Flooding, Distance Vector, Link State, Hierarchical, Broadcast, Multicast, Anycast, Congestion Control Algorithms, Quality of Service.		
At the end of the Module 3, students will be able to: 1. Understand design issues of network layer. (BL-2) 2. Explain efficient routing protocols in computer networks. (BL-2) 3. Describe elements of network layer required for data transfer over Internet. (BL-2)		
MODULE – 4	**Transport Layer**	**(10H)**
Internetworking, The network layer in the Internet: IPV4 Addresses, IPV6, Internet Control protocol, BGP. The Transport Layer: The Transport layer services, Elements of Transport Protocols, The Internet transport protocols: UDP, TCP., Sliding Window Protocols,		
At the end of the Module 4, students will be able to: 1. Understand the services provided by transport layer. (BL-2) 2. Describe elements of transport layer required for data transfer over Internet. (BL-2) 3. Demonstrate end to end communication. (BL-3) 4. Discuss performance issues in Transport Layer(BL-2)		

MODULE – 5	Application Layer	(9H)
Application Layer: Introduction, World Wide Web and HTTP, Domain Name System, FTP, e-mail, TELNET, Secure Shell.		
At the end of the Module 5, students will be able to: 1. Explain the working of world wide web with HTTP, DNS. (BL-2) 2. Describe the protocols for mail, remote system login. (BL-2) 3. Discuss file transfer, network management protocols. (BL-2)		
	Total hours:	**48 hours**

Content beyond syllabus:

1. Connecting Devices and VPN
2. Peer-to-Peer paradigm

Text Book(s):

3. Behrouz A. Forouzan, Data communications and networking, 5th edition, Mc Graw Hill Education, 2012.
4. Andrew S. Tanenbaum, Wetherall, Computer Networks, 5th edition, Pearson, 2013.

Reference Book(s):

8. Douglas E. Comer, Internetworking with TCP/IP – Principles, protocolsand architecture-Volume 15th edition, PHI.
9. Kurose James, Ross Keith, Computer Networking: A Top-Down Approach, 6th Edition, Pearson Education.
10. Behrouz A. Forouzan, TCP/IP Protocol Suite, 4th edition, Tata McGraw Hill

NARAYANA ENGINEERING COLLEGE:GUDUR								
	MOBILE APPLICATION DEVELOPMENT							R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
21CS3008	3	0	0	48	3	40	60	100

Pre-requisite: Java programming and Object-oriented programming, Basics of any Scripting Language.

Course Objectives:

1. To understand fundamentals of android operating systems.
2. To understand the platform, tools, technology and process for developing mobile applications.
3. To demonstrate the operation of the application, configuration files, intents and activities.
4. To develop and deploy Android applications.
5. To illustrate the various components, layouts and views in creating android applications.

Course Outcomes: After successful completion of the course, student will be able to:

CO 1	Identify a significant programming component, involving the sensors and hardware features of mobile device. (BL-2)
CO 2	Demonstrate the use of Android software development controls. (BL-2)
CO 3	Construct mobile applications on the Android Platform using different layouts for playing video and audio. (BL-3)
CO 4	Acquire the Information Using Dialogs and Fragments by the mobile applications for the Android operating system. (BL-3)
CO 5	Prepare mobile applications involving Menus and Action Bars. (BL-3)

CO-PO Mapping														
CO	**PO**												**PSO**	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	3	1	1										1	1
CO2	2	1	2	1									2	2
CO3	2	2	2	2	2								2	1
CO4	1	1	2	2								1	1	2
CO5	2	3	3	1								1	2	1
1: Low, 2-Medium, 3- High														

COURSE CONTENT		
MODULE – 1	**Introduction to Android**	**12H**
The Android 4.1 jelly Bean SDK, Understanding the Android Software Stack, installing the Android SDK, Creating Android Virtual Devices, Creating the First Android Project, Using the Text view Control, Using the Android Emulator, The Android Debug Bridge(ADB), Launching Android Applications on a Handset.		
At the end of the Module 1, students will be able to: 1. Observe the features of android software. (BL-2) 2. Understand the order of Android software stack. (BL-2) 3. Discover and Launch an android application on a handset. (BL-2)		
MODULE -2	**Basic Widgets**	**10H**
The Role of Android Application Components, Utility of Android API, Overview of the Android Project Files, Understanding Activities, Role of the Android Manifest File, Creating the User Interface, Commonly Used Layouts and Controls, Event Handling, Displaying Messages Through Toast, Creating and Starting an Activity, Using the Edit Text Control, Choosing Options with Checkbox, Choosing Mutually Exclusive Items Using Radio Buttons.		
At the end of the Module 2, students will be able to: 1. Differentiate the hierarchy of files and sub files. (BL-2) 2. Understand the importance of Manifest file. (BL-2) 3. Select the widgets and group different controls for event handling. (BL-2)		

MODULE-3	Building Blocks for Android Application Design	9H
Introduction to Layouts, Linear Layout, Relative Layout, Absolute Layout, Using Image View, Frame Layout, Table Layout, Grid Layout, Adapting to Screen orientation. Utilizing Resources and Media Resources, Creating Values Resources, Using Drawable Resources, Switching States with Toggle Buttons, Creating an Images Switcher Application, Scrolling Through Scroll View, playing Audio, Playing Video		
At the end of the Module 3, students will be able to: 1. Construct an android application using layouts. (BL-3) 2. Operate audio and video on hand set. (BL-3) 3. Apply displaying progress with Scrolling Through Scroll View. (BL-3)		
MODULE-4	**Selection widgets And Fetching Information Using Dialogs and Fragments**	**9H**
Using List View, Using the Spinner control, Using the GridView Control, Creating an Image Gallery Using the ViewPager Control. Dialogs, Selecting the Date and Time in One Application, Fragments, Creating Special Fragments**.**		
At the end of the Module 4, students will be able to: 1. Choose and select which one is the best view of list. (BL-3) 2. Develop customized dialogs. (BL-3) 3. Selecting the Date and Time in an Application.(BL-3)		
MODULE-5	**Building Menus**	**8H**
Creating Interface Menus and Action Bars, Menus and Their Types, Creating Menus Through XML, Creating Menus Through Coding, Applying a Context Menu to a List View, Using the Action Bar, Replacing a Menu with the Action Bar, Creating a Tabbed Action Bar, Creating a Drop-Down List Action Bar.		
At the end of the Module 5, students will be able to: 1. Prepare and produce information through menus. (BL-3) 2. Visualize the Action Bar. (BL-3) 3. Manipulate a Menu with the Action Bar. (BL-3)		
	Total hours:	**48 hours**

Content beyond syllabus: Advanced Android Programming: Gaming engines like Unity, Unreal Engine Etc..

Text Book(s):

1. B.M Harwani, Android Programming, Pearson Education.
2. Lauren Darcey and Shane Conder, "Android Wireless Application Development", 2nd edition, Pearson Education.

Reference Book(s):

1. Professional Android Application Development, Wiley India Private Limited.
2. Dawn Griffiths, David Griffiths, "Head First Android Development: A Brain-Friendly Guide", Second Edition, O'Reilly Media, 2017.
3. James C Sheusi, Android application Development for Java Programmers, Cengage Learning.
4. w.FrankAbleson, Robi Sen, Chris King, C.Enrique Ortiz., Android In Action,Dreamtech.
5. RetoMeier,Professional Android 4 applications development, Wiley India.
6. Wei- Meng Lee, Beginning Android 4 applications development, Wiley India.

NARAYANA ENGINEERING COLLEGE::GUDUR								
21CS3009	**WEB TECHNOLOGIES**							R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
21CS3009	3	0	0	48	3	40	60	100

Pre-requisite: Knowledge of Information Technology

Course Objectives:

1. To impart basic web application development skills.
2. To translate user requirements into the overall architecture and implementation of new systems and manage project and coordinate with the client.
3. To develop scripting code in PHP language and Writing optimized front end code HTML and JavaScript.
4. To create and debug database related queries and Create test code to validate the applications against client requirement.
5. To monitor the performance of web applications, infrastructure and Troubleshooting web applications with a fast and accurate resolution.

Course Outcomes: On successful completion of the course, the student will be able to:

CO 1	Construct static web pages using HTML and CSS. **(BL-3)**
CO 2	Implement various concepts related to dynamic web pages and validate them using JavaScript**.** **(BL-3)**
CO 3	Create secure, usable database driven web applications**.** **(BL-3)**
CO 4	Develop web Applications using Scripting Languages. **(BL-3)**
CO 5	Explain the concepts of Extensible Mark-up Language **(BL-2)**

CO-PO Mapping														
CO	**PO**												**PSO**	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	1	2	2										1	2
CO2	2	3	3	1									1	2
CO3	2	3	3	1									1	2
CO4	1	2	3	1									1	2
CO5	2	2	3										1	1
1: Low, 2-Medium, 3- High														

COURSE CONTENT		
MODULE - 1	**HTML, CSS &Web Servers**	**(10H)**
HTML: Basic Syntax, Standard HTML Document Structure, Basic Text Mark-up, HTML styles, Elements, Attributes, Heading, Layouts, HTML media, Iframes Images, Hypertext Links, Lists, Tables, Forms, GET and POST method, HTML 5, Dynamic HTML. Cascading style sheets, Levels of Style Sheets, Style Specification Formats, Selector Forms, The Box Model, Conflict Resolution, CSS3, Web Servers- Apache, IIS, Bundle Servers.		
At the end of the Module 1, students will be able to: 1. Understand the basics of web programming. (BL-2) 2. Explain tags in HTML, CSS. (BL-2) 3. Construct static web pages using HTML tags. (BL-3) 4. Install and configure web servers, bundle servers. (BL-3)		
MODULE - 2	**Java Script**	**(10 H)**
Java script: Introduction to Java script, Objects, Primitives Operations and Expressions, Control Statements, Arrays, Functions, Constructors, Pattern Matching using Regular Expressions, Exception Handling, Validation, Built-in objects, Event Handling, DHTML with JavaScript., DOM Model		

At the end of the Module 2, students will be able to:

1. Explain basic programming constructs of java script. (BL-2)
2. Develop dynamic and interactive web pages. (BL-3)
3. Perform validations for the web pages. (BL-2)

MODULE - 3	**PHP**	**(9 H)**

PHP Data types and Concepts: The anatomy of a PHP Page, Variables and data types, Operators, Expressions and Statements, Strings, Arrays and Functions.

At the end of the Module 3, students will be able to:

1. Compare java and php programming features. (BL-2)
2. Understand the anatomy of php page. (BL-2)
3. Explain various PHP programming constructs. (BL-2)
4. Implement simple PHP programs in the server. (BL-3)

MODULE - 4	**PHP Advanced Concepts**	**(9 H)**

PHP Advanced Concepts: UsingCookies, Using HTTP Headers, Using Sessions, authenticating users, Using Environment and Configuration variables, Working with Date and Time.

At the end of the Module 4, students will be able to:

1. Understand cookies, http headers, sessions. (BL-2)
2. Explain user authentication in PHP. (BL-2)
3. Analyze PHP document structure. (BL-3)

MODULE - 5	**Extensible Markup Language**	**(10 H)**

Working with XML: Document type Definition (DTD), XML schemas, XSLT, Document object model, Parsers - DOM and SAX. News Feed (RSS and ATOM). Java Web Services: Web services Basics – Creating, Publishing, Testing and Describing a Web services (WSDL)-Consuming a web service, SOAP.

At the end of the Module 5, students will be able to:

1. Understand the structure of Document type Definition (DTD), XML schemas. (BL-2)
2. Analyze parsing of XML document with DOM, SAX. (BL-3)
3. Demonstrate web service with SOAP, WSDL in Java web application development. (BL-2)

Total hours:	**48 Hours**

Text Book(s):

1. Robet W Sebesta, Programming the World Wide Web, 7th Edition, Pearson, 2013
2. Uttam K Roy, Web Technologies, 1stEdition ,7th impression, Oxford, 2012
3. Lee Babin, Nathan A Good, Frank M. Kromann and Jon Stephens, PHP 5 Recipes A problem Solution Approach.

Reference Book(s):

8. Deitel and Deitel and Nieto, Internet and World Wide Web - How to Program, , 5th Edition, Prentice Hall, 2011.
9. ELad Elrom, Pro Mean Stack Development, 1st Edition, Apress O'Reilly, 2016
10. David sawyer mcfarland, Java Script & jQuery the missing manual, 2nd Edition, O'Reilly, 2011
11. Peter Pollock, Web Hosting for Dummies, 1st Edition, John Wiley & Sons, 2013
12. Tom Christiansen, Jonathan Orwant, Programming Perl, 4th Edition, O'Reilly, 2012
13. Kogent L S, Web Technologies: HTML, JavaScript, PHP, Java, JSP, XML and AJAX, Black book, 1st Edition, Dream Tech, 2009
14. Paul S Wang, Sanda S Katila, An Introduction to Web Design, Programming, 1st Edition, Cengage Learning, 2003

Virtual Lab:

List editors which can be used to create HTML documents.

Understand: Describe the Structure of HTML document.

Apply: Identity different Tags are given in HTML.

Analyze: Compare the various HTML Tags.

1. Introduction to HTML
2. Applying Attributes in HTML Tags
3. Inserting images through img tags
4. Using Anchor Tags for Hyperlinks
5. How marquee Tags work in HTML
6. Creating Tables in HTML
7. Types of Lists in HTML
8. Working of div Tag in HTML
9. Embedding through iframe Tag
10. Creating Webpage Layout in HTML

NARAYANA ENGINEERING COLLEGE::GUDUR								
	ARTIFICIAL INTELLIGENCE							R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
21CS3010	3	0	0	48	3	40	60	100

Pre-requisite:

Mathematical Foundations of Computer Science, Computer Programming, Data Structures and Algorithms.

Course Objectives:

1. To understand the importance of the task environment in determining the appropriate agent design.
2. To teach the concepts of state space representation, heuristic search together with the time and space complexities
3. To describe the various types of learning methods and natural language processing.
4. To provide basic knowledge on natural language for communication and perception.
5. To understand the basic knowledge on robotics and philosophical foundations of AI.

Course Outcomes: On successful completion of the course, student will be able to:

CO 1	Understand the role of agents, environments and relationship among them.(BL-2)
CO 2	Examine various problem-solving approaches in searching and learning. (BL-2)
CO 3	Demonstrate the use of Reinforcement learning and natural language processing.(BL-3)
CO 4	Understand the natural language for communication and object perception (BL-2)
CO 5	Demonstrate the role of Robot in various applications and list out philosophical issues in AI. (BL-2)

CO-PO Mapping														
CO	**PO**												**PSO**	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	1	1											1	
CO2	2	3											1	
CO3	2	3											1	
CO4	3	3											1	
CO5	3	2											1	
1: Low, 2-Medium, 3- High														

COURSE CONTENT		
MODULE – 1	**Introduction to Artificial Intelligence**	**10H**
Introduction: AI Definition, Foundations of Artificial Intelligence, History of Artificial Intelligence. Intelligent Agents: Agents and Environments, Good Behavior Concept of Rationality, Nature of Environments, The Structure of Agents. Problem-Solving Agents, Searching for Solutions; Uninformed Search Strategies: Breadth-first search, Uniform-cost search, DFS: Informed (Heuristic) Search strategies: Greedy BFS, A^* search.		
At the end of the Module 1, students will be able to: 1. Understand the basics and applications of Artificial intelligence.(BL-2) 2. Illustrate how rationality can be applied to a wide variety of agents.(BL-2) 3. Demonstrate the various search strategies and heuristics. (BL-2)		
MODULE – 2	**Problem Solving beyond classical search and Learning**	**10H**
Local search algorithms and optimization problems**:** Hill-climbing, simulated annealing; Local Search in Continuous Spaces, Searching with Non-Deterministic Actions, Searching with partial observations, Online Search Agents and Unknown Environment. Forms of Learning, Supervised Learning, Learning Decision Trees, Logical Formulation of Learning, Explanation-Based Learning, Learning Using Relevance Information, Inductive Logic Programming.		
At the end of the Module 2, students will be able to: 1. Understand advanced classical searching Techniques.(BL-2) 2. Demonstrate Online Search Agents, Non-Deterministic Actions & Partial		

Observations.(BL-2)

3. Gain knowledge on basic forms of learning, learning decision trees and Explanation-based learning (BL-2)

MODULE – 3	Reinforcement Learning and Natural LanguageProcessing	10H

Introduction, Passive Reinforcement Learning, Active reinforcement Learning, Generalization in Reinforcement Learning, Policy Search, applications of Reinforcement Learning**,** Language Models, Text Classification, Information Retrieval, Information Extraction.

At the end of the Module 3, students will be able to:

1. Understand the Reinforcement learning methods and policy search. (BL-2)
2. Demonstrate language models and text classification. (BL-3)
3. Gain knowledge on Information retrieval and extraction. (BL-2)

MODULE – 4	Natural Language for communication and Perception	9H

Phrase structure grammars, Syntactic analysis, Augmented grammars and semantic Interpretation, Machine translation, Speech Recognition. Image formation, Early Image Processing Operations, Object recognition by appearance, Reconstructing the 3D World, Object recognition from structural information, Using Vision**.**

At the end of the Module 4, students will be able to:

1. Understand Syntactic analysis and semantic interpretation.(BL-3)
2. Demonstrate machine translation and speech recognition.(BL-3)
3. Gain knowledge on Object recognition and how to use Vision(BL-2)

MODULE – 5	Robotics and Philosophical foundations	9H

Introduction, Robotic Hardware, Robotic Perception, Planning to move, Planning uncertain movements, Moving, Robotic software architectures, application domains.

Week AI, Strong AI, Ethics and Risks of AI, Agent Components and Agent architectures, Are we going in the right direction, What if AI does succeed.

At the end of the Module 5, students will be able to:

1. Understand the basics of robotics. (BL-2)
2. Demonstrate robotic hardware, software and applications. (BL-2)
3. Understand the philosophical foundations and agent architectures.(BL-2)

Total hours:	**48 hours**

Content beyond syllabus:

1. Constraint Satisfaction Problems.
2. Planning
3. Uncertain Knowledge and reasoning

Text Book(s):

1. Stuart Russell and Peter Norvig, Artificial Intelligence A Modern Approach, 3rdEdition, Pearson Education.
2. Elaine Rich, Kevin Knight & Shivashankar B Nair, "Artificial Intelligence", 3rd Edition, McGraw Hill Education.

Reference Book(s):

1. Patrick Henny Winston, Artificial Intelligence, 3rdEdition, Pearson Education.
2. Patterson, Introduction to Artificial Intelligence and Expert Systems, 1stEdition Pearson India.
3. George F Lugar, Artificial intelligence, structures and Strategies for Complex problem solving,6thed, PEA, 2008
4. Poole, D. and Mackworth,Artificial Intelligence: Foundations of Computational Agents,,Cambridge University Press. 2010
5. Padhy, N.P ,Artificial Intelligence and Intelligent Systems,. 2009,Oxford University Press.

NARAYANA ENGINEERING COLLEGE::GUDUR							
	CRYPTOGRAPHY AND NETWORK SECURITY						R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks	
	L	T	P		3	CIE	SEE	TOTAL
21CS3011	3	0	0	50		40	60	100

Pre-requisite:

1. Knowledge on Computer Networks and Data Communication.
2. Knowledge on Information Security.

Course Objectives:

1. Introduce the basic categories of threats to computers and networks
2. Illustrate various cryptographic algorithms.
3. Demonstrate public-key cryptosystem.
4. Discuss the fundamental ideas of public-key cryptography.
5. Explore Web security threats and protection mechanisms

Course Outcomes: After successful completion of the course, student will be able to:

CO 1	Understand and apply the cryptographic algorithms to safeguard from intruders(BL-2,3)
CO 2	Compare and contrast symmetric and asymmetric encryption systems and their vulnerability to attack(BL-4)
CO 3	Implement the various key distribution, management and message authentication schemes to send the messages with security(BL-3)
CO 4	Identify information system requirements for Transport level, wireless network, E-Mail and IP(BL-2)
CO 5	Design a network security system by implementing all the concepts of encryption and decryption algorithms(BL-6)

CO-PO Mapping

CO	PO												PSO	
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO 2**
CO 1	**3**	**2**						**1**					**2**	
CO 2	**3**	**3**	**3**										**3**	
CO 3	**3**	**3**	**1**										**1**	
CO 4	**3**	**2**	**3**					**1**					**1**	
CO 5	**3**	**3**	**1**					**2**					**2**	

1: Low, 2-Medium, 3- High

COURSE CONTENT		
MODULE – 1		**8H**

Attacks on Computers and Computer Security: Introduction, The need for security, Principles of security, Types of Security attacks, Security services, Security Mechanisms, A model for Network Security Cryptography, plain text and cipher text, substitution techniques, transposition techniques, encryption and decryption, symmetric and asymmetric key cryptography, Steganography.

LEARNING OUTCOMES:

At the end of 1 Module students will be able:

1. Identify different types of Attacks (L3)
2. Interpret various cryptography techniques (L5)
3. Distinguish between cryptography and Steganography (L4)

MODULE – 2		**9H**

Symmetric key Ciphers: Block Cipher principles &Algorithms (DES, AES, Blowfish), Block cipher modes of operation, Stream ciphers, Key distribution.

Asymmetric key Ciphers: Principles of public key cryptosystems, Algorithms (RSA, Diffie Hellman, ECC), Key Distribution.

LEARNING OUTCOMES:

At the end of this Module students will be able:

1. Differentiate symmetric and asymmetric ciphers (L4)
2. Explain the principles of public key cryptography (L2)
3. Select the appropriate cryptographic algorithm based on the requirements and applications.(L5)

MODULE – 3		**12H**

Message Authentication Algorithms and Hash Functions: Authentication requirements, Functions, Message authentication codes, Hash Functions, Secure hash algorithm, Whirlpool, HMAC, CMAC, Digital signatures, knapsack algorithm.

LEARNING OUTCOMES:

At the end of this Module students will be able:

1. Summarize authentication techniques (L2)
2. Apply Hash algorithm for generating Digital signatures (L3)

MODULE – 4		**9H**

E-Mail Security: Pretty Good Privacy, S/MIME IP Security: IP Security overview, IP Security architecture, Authentication Header, Encapsulating security payload, security associations, key- management.

LEARNING OUTCOMES:

At the end of this Module students will be able:

1. Extend security for emails (L2)
2. Examine IP security mechanisms (L4)

MODULE – 5		**10H**

Web Security: Web security considerations, Secure Socket Layer and Transport Layer Security, Secure electronic transaction Intruders, Virus and Firewalls: Intruders, Intrusion detection, password management, Virus and related threats, Firewall design principles, Types of firewalls Case Studies on Cryptography and security: Secure Inter-branch Payment Transactions, Cross site Scripting Vulnerability, Virtual Elections

LEARNING OUTCOMES:

At the end of this Module students will be able:

1. Design secure electronic transactions (L6)
2. Explain different types of Firewalls (L2)

Total hours:	**48 hours**

Text Book(s):

1. William Stallings, "Cryptography and Network Security", 5th Edition, PearsonEducation, 2011.
2. Bernard Menezes "Network Security and Cryptography", 1stEdition, CENGAGE Learning, 2010.

Reference Book(s):

1. C K Shyamala, N Harini, Dr T R Padmanabhan, Wiley India, "Cryptography and Network Security",1st Edition, Wiley India Pvt Ltd, 2011.
2. Forouzan Mukhopadhyay "Cryptography and Network Security", 2nd Edition , McGraw Hill, 2010.
3. Mark Stamp, Wiley India, "Information Security, Principles and Practice", 2nd Edition, Wiley, 2011

NARAYANA ENGINEERING COLLEGE::GUDUR							
	CLOUD COMPUTING						R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks	
	L	T	P		C	CIE	SEE	TOTAL
21CS3012	3	0	0	50	3	40	60	100

Course Outcomes: After successful completion of the course, student will be able to:	
CO 1	Summarize the basic concepts of, Cloud technologies for development of Cloud applications (BL-2)
CO 2	Develop cloud Applications through Cloud Technologies(BL-3)
CO 3	Interpret Cloud service architectures in Cloud environment(BL-3)
CO 4	Analyse the core issues of cloud computing. (BL-3)
CO 5	Choose appropriate technologies, algorithms and approaches to used in cloud Computing(BL-3)

CO-PO Mapping														
CO	**PO**												**PSO**	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO1**	**PSO 2**
CO1	1	1											1	
CO2	3	1											1	
CO3	1	2											2	1
CO4	2	1	2										1	1
CO5	1	1	1										1	
1: Low, 2-Medium, 3- High														

COURSE CONTENT		
MODULE – 1		**9H**
Cloud Computing Insights- Distributed Computing, High Performance Computing, Utility and Enterprise Grid Computing, Cluster Computing**,** Cloud Computing fundamentals, Essential Characteristics, On Demand Self Service, Location independent resource pooling, Elastic Computing, Measured Service, Comparing cloud providers with traditional IT service providers, Vendor Lock-in, security level of third party- Security issues: Government policies.		
At the end of the Module 1, students will be able to: 1. Outline the Cloud characteristics and models.(BL-2) 2. understand security issues in cloud computing(BL-2)		
MODULE – 2		**10H**
Cloud computing architecture, Layers of Cloud computing- IaaS, PaaS and SaaS, Cloud deployment models- Private, Public, Hybrid and Community Clouds, Advantages of Cloud Computing.		
At the end of the Module 2, students will be able to: 1. Design and build cloud applications.(BL-6) 2. Describe the multimedia cloud. (BL-2)		
MODULE – 3		**10H**
Introduction, Characteristics of Virtualized Environments, Virtualization and Cloud Computing, Pros and Cons of Virtualization, Virtual machines and Virtualization of Clusters and Data Centres, Case studies – Xen Virtual Machine monitors – Xen API, VMware- VMware products- VMware features, Microsoft Virtual Server- Features of Microsoft Virtual Server, Open stack.		
At the end of the Module 3, students will be able to: 1. Classify different models, different technologies in cloud.(BL-2) 2. Understand Microsoft virtual server concepts(BL-2)		
MODULE – 4		**10H**
Cloudsim Open source framework, Simulate VMs, memory, network, disks; Aneka – Cloud computing Framework for Enterprise Cloud applications development, Aneka Architecture, Programming models: Thread, Task and Map Reduce		
At the end of the Module 4, students will be able to: 1. Illustrate applications of cloud computing 2. Apply cloud computing concepts using programming models		
MODULE – 5		**10H**
Case studies – Salesforce.com for SaaS application development, GAE- Google App Engine, Microsoft Windows Azure – public resources for VMs and Services, AWS- Amazon Web Services – public cloud registration, Services, OpenStack – Open Source Development		

Platform for Clouds and tools.	
At the end of the Module 5, students will be able to: 1. Understand Cloud computing and Virtualization.(BL-1) 2. Deploying SaaS application on Google App engine or Azure cloud.(BL-3)	
Total hours:	**49 hours**

TEXTBOOK:

1. RajkumarBuyya, Christian Vecchiola, S. ThammaraiSelvi, "Mastering Cloud Computing – Foundations and applications", McGraw Hill Publications,
2. Toby Velte, Anthony Velte, Robert Elsenpeter, "Cloud Computing – A Practical Approach", Mc Graw Hill, Inc, New York, NY, USA.

REFERENCES:

1. Kai Hwang, Geoffrey C Fox, Jack J. Dongarra, "Distributed and Cloud Computing, Morgan Kaufmann.
2. Cloud Computing Principles and Paradigms, John Wiley & Sons publications

THE PROFESSIONAL ELECTIVES

The Professional Elective Courses (PE) are shown in different tracks/groups: The students will have options of selecting the electives from the different tracks/groups depending on the specialization one wishes to acquire.

Electives Track/ Groups	Professional Elective-1	Professional Elective-2	Professional Elective-3	Professional Elective-4	Professional Elective-5
Computer Networks and Securities	Wireless Sensor Networks 21CS4001	Ethical Hacking 21CS4006	Information and Cyber Security 21CS4011	Computer Forensics 21CS4016	Block chain Technologies 21CS4021
Software Engineering	Software Project Management 21CS4002	Software Architecture 21CS4007	Software Testing 21CS4012	Object Oriented Analysis and Design 21CS4017	DEVOPS 21CS4022
Data Science and Engineering	Data warehousing and data mining 21CS4003	Business Intelligence and Analytics 21CS4008	**Data Virtualization Techniques** 21CS4013	**Reinforcement Learning** 21CS4018	Tools and Techniques for Data Science 21CS4023
Cloud Computing	Distributed Systems 21CS4004	Service Oriented Architecture 21CS4009	Cloud Computing 21CS4014	High Performance Computing 21CS4019	Cloud Security 21CS4024
Virtualization and Others	Game Development 21CS4005	**Big Data Analytics** 21CS4010	**Information Storage and Retrieval Systems** 21CS4015	Augmented and Virtual Reality 21CS4020	Virtualization Technologies 21CS4025
MOOCS	**MOOCS-1 21CS4026**	**MOOCS-2 21CS4027**	**MOOCS-3 21CS4028**	**MOOCS-4 21CS4029**	**MOOCS-5 21CS4030**

Professional Elective-1

NARAYANA ENGINEERING COLLEGE:GUDUR							
21CS4001	WIRELESS SENSOR NETWORKS						R21
	Hours / Week			Total hrs	Credit	Max Marks	
	L	T	P		C	CIE	SEE	TOTAL
	3	0	0	48	3	40	60	100

Pre-requisite: Basic knowledge of Data Communication Networks

Course Objectives:

1. To make students understand the basics of Wireless sensor Networks.
2. To familiarize with learning of the Architecture of WSN.
3. To understand the concepts of Networking and Networking in WSN.
4. To study the design consideration of topology control and solution to the various problems.
5. To introduce the hardware and software platforms and tool in WSN.

Course Outcomes: After successful completion of the course, Student will be able to:

CO 1	Understand challenges and technologies for wireless networks
CO 2	Understand architecture and sensors
CO 3	Describe the communication, energy efficiency, computing, storage and transmission
CO 4	Establishing infrastructure and simulations
CO 5	Explain the concept of programming the in WSN environment

CO-PO Mapping

CO	PO												PSO	
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
CO1	3	1											1	
CO2	3	3											2	
CO3	3	2											2	
CO4	2	2	2										1	1
CO5	2	2	1										1	

1-Low, 2- Medium, 3- High

COURSE CONTENT		
MODULE – 1	**Introduction to Sensor Networks**	**10H**
Introduction to Sensor Networks, unique constraints and challenges, Advantage of Sensor Networks, Applications of Sensor Networks, Types of wireless sensor networks, IEEE 802.11		
MODULE – 2	**MANETS and Sensor Networks**	**10H**
Mobile Ad-hoc Networks (MANETs) and Wireless Sensor Networks, Enabling technologies for Wireless Sensor Networks. Issues and challenges in wireless sensor networks		
MODULE – 3	**Routing Protocols and MAC Protocols**	**10H**
Routing protocols: Energy-Efficient Routing, Geographic Routing, MAC protocols: Classification of MAC Protocols, S-MAC Protocol, B-MAC protocol, IEEE 802.15.4 standard and ZigBee		
MODULE – 4	**Data Dissemination and other concepts**	**(9H)**
Dissemination protocol for large sensor network. Data dissemination, data gathering, and data fusion; Quality of a sensor network; Real-time traffic support and security protocols, Design Principles for WSNs		
MODULE – 5	**WSN and Internet Communication**	**9H**
WSN to Internet Communication, and Internet to WSN Communication. Single-node architecture, Hardware components & design constraints, Operating systems and execution environments, introduction to Tiny OS and nesC.		
Total hours:		**48 hours**

TEXTBOOK:

1. Ad-Hoc Wireless Sensor Networks- C. Siva Ram Murthy, B. S. Manoj, Pearson
2. Principles of Wireless Networks – Kaveh Pah Laven and P. Krishna Murthy, 2002, PE

REFERENCES:

1. Wireless Digital Communications – Kamilo Feher, 1999, PHI.
2. Wireless Communications-Andrea Goldsmith, 2005 Cambridge University Press.
3. Mobile Cellular Communication – Gottapu Sasi bhushana Rao, Pearson Education, 2012.
4. Wireless Communication and Networking – William Stallings, 2003, PHI

NARAYANA ENGINEERING COLLEGE:GUDUR								
	SOFTWARE PROJECT MANAGEMENT							R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
21CS4002	3	0	0	48	3	40	60	100

Pre-requisite:Software Engineering

Course Objectives:

1Understanding the specific roles within a software organization as related to project and process management.

6. Understanding the basic infrastructures competences (e.g., process modeling and
7. Describe the principles, techniques, methods & tools for model based management of software projects, assurance of product quality and process
8. Understanding the basic steps of project planning, project management, quality assurance, and process management and their relationships.
9. To provide basic project management skills with a strong emphasis on issues and problems associated with delivering successful IT projects.

Course Outcomes: After successful completion of the course, Student will be able to:

CO 1	Identify the concepts of conventional software project management and Software Economics for developing a software project.
CO 2	Apply Conventional and modern principles of software project management to develop the software products.
CO 3	Explain the software architecture, life cycle phases and process for a building a software product.
CO 4	Interpret the techniques to evaluate progress of software project workflows in terms of milestones and check points, project organization responsibilities and process automation
CO 5	Choose the software metrics to implement a software product through process instrumentation ethical principles to be followed in management of software economics

CO-PO Mapping

CO	PO												PSO	
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2
CO1	3	3									1		2	3
CO2	2	2	2								3	2		
CO3	2	2	2								1		3	2
CO4	2	3	2								3	1	2	3
CO5	2	2	3								3		2	2
1-Low, 2- Medium, 3- High														

COURSE CONTENT		
MODULE – 1	**Conventional Software Management**	**10H**
The waterfall model, conventional software Management performance. Evolution of Software Economics: Software Economics, pragmatic software cost estimation		
MODULE – 2	**Improving Software Economics**	**10H**
Reducing Software product size, improving software processes, improving team effectiveness, improving automation, Achieving required quality, peer inspections **The old way and the new**: The principles of conventional software engineering, principles of modern software management, transitioning to an iterative process		
MODULE – 3	**Life cycle phases**	**10H**
Engineering and production stages, inception, Elaboration, construction, transition phases. Artifacts of the process: The artifact sets, Management artifacts, Engineering artifacts, programmatic artifacts. Model based software architectures: A Management perspective and technical perspective		
MODULE – 4	**Work Flows of the process**	**(9H)**
Software process workflows, Inter Trans workflows. Checkpoints of the Process: Major Mile Stones, Minor Milestones, Periodic status assessments. Iterative Process Planning: Work breakdown structures, planning guidelines, cost and schedule estimating, Interaction planning process, Pragmatic planning. **Project Organizations and Responsibilities**: Line-of-Business Organizations, Project Organizations, evolution of Organizations. Process Automation: Automation Building Blocks, The Project Environment		
MODULE – 5	**Project Control and Process instrumentation**	**9H**
The server care Metrics, Management indicators, quality indicators, life cycle expectations pragmatic Software Metrics, Metrics automation. Tailoring the Process: Process discriminates, Example. **Future Software Project Management**: Modern Project Profiles Next generation Software economics, modern Process transitions		
	Total hours:	**48 hours**

TEXTBOOK:

Bob Hughes, Mike Cotterell and Rajib Mall: Software Project Management – FifthEdition, Tata McGraw Hill, New Delhi, 2012.

REFERENCES:

1. Robert K. Wysocki "Effective Software Project Management" – Wiley Publication, 2011.
2. Walker Royce: "Software Project Management"- Addison-Wesley, 1998.
Gopala swamy Ramesh, "Managing Global Software Projects" – McGraw HillEducation (India), Fourteenth Reprint 2013.

NARAYANA ENGINEERING COLLEGE:GUDUR							
	DATA WAREHOUSING AND DATA MINING						R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks	
	L	T	P		C	CIE	SEE	TOTAL
21CS4003	3	0	0	48	3	40	60	100

Pre-requisite: Programming Skills

Course Objectives:

1. Understand the basic concepts of data warehousing.
2. Recognize the characteristics of OLAP systems.
3. Understand the knowledge discovery process..

Course Outcomes: After successful completion of the course, Student will be able to:

CO 1	Designing, building, and using data warehouses for decision support.
CO 2	Have a solid foundation in the principles, techniques, and applications of data mining systems, enabling them to apply these skills to extract meaningful insights from diverse datasets.
CO 3	To effectively mine and evaluate frequent patterns, associations, and correlations, and to apply these patterns in practical scenarios, particularly in the context of classification tasks.
CO 4	Understanding of classification and clustering techniques, along with the ability to apply these methods to real-world problems, evaluate their performance, and improve their effectiveness.
CO 5	To effectively use WEKA for data analysis, apply machine learning algorithms to real-world datasets, and interpret the results to make informed decisions.

CO-PO Mapping

CO	PO												PSO	
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
CO1	3	3	2	1									3	2
CO2	2	3	3	2									2	2
CO3	3	3	3	1									2	2
CO4	2	3	3	1									2	1
CO5	2	3	3	2									2	2

1-Low, 2- Medium, 3- High

COURSE CONTENT		
MODULE – 1		**10H**
Basic Concepts – Data Warehousing Components – Building a Data Warehouse – Database Architectures for Parallel Processing – Parallel DBMS Vendors – Multidimensional Data Model Data Warehouse Schemas for Decision Support, Concept Hierarchies Characteristics of OLAP Systems – Typical OLAP Operations, OLAP and OLTP.		
MODULE – 2		**9H**
Introduction to Data Mining Systems – Knowledge Discovery Process – Data Mining Techniques – Issues – applications- Data Objects and attribute types, Statistical description of data, Data Preprocessing – Cleaning, Integration, Reduction, Transformation and discretization, Data Visualization, Data similarity and dissimilarity measures.		
MODULE – 3		**9H**
Mining Frequent Patterns, Associations and Correlations – Mining Methods- Pattern Evaluation Method – Pattern Mining in Multilevel, Multi Dimensional Space – Constraint Based Frequent Pattern Mining, Classification using Frequent Patterns.		
MODULE – 4		**10H**
Decision Tree Induction – Bayesian Classification – Rule Based Classification – Classification by Back Propagation – Support Vector Machines — Lazy Learners – Model Evaluation and Selection-Techniques to improve Classification Accuracy. Clustering Techniques – Cluster analysis-Partitioning Methods – Hierarchical Methods – Density Based Methods – Grid Based Methods – Evaluation of clustering – Clustering high dimensional data-Clustering with constraints, Outlier analysis-outlier detection methods.		
MODULE – 5		**10H**
Datasets – Introduction, Iris plants database, Breast cancer database, Auto imports database – Introduction to WEKA, The Explorer – Getting started, Exploring the explorer, Learning algorithms, Clustering algorithms, Association–rule learners.techniques.		
	Total hours:	**48 hours**

TEXTBOOK:

.Jiawei Han and Micheline Kamber, ―Data Mining Concepts and Techniques, Third Edition, Elsevier, 2012.

REFERENCES:

1. Alex Berson and Stephen J. Smith, ―Data Warehousing, Data Mining & OLAP‖, TataMcGraw – Hill Edition, 35th Reprint 2016.

2. K.P. Soman, Shyam Diwakar and V. Ajay, ―Insight into Data Mining Theory and Practice,Eastern Economy Edition, Prentice Hall of India, 2006.

3. Ian H.Witten and Eibe Frank, ―Data Mining: Practical Machine Learning Tools and Techniques, Elsevier, Second Edition.

NARAYANA ENGINEERING COLLEGE: GUDUR								
	DISTRIBUTED SYSTEMS							R21
Course Code	Hours/Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
21CS4004	3	0	0	50	3	40	60	100

Pre-requisite: Operating systems knowledge can be helpful when learning about distributed systems, as distributed systems often rely on concepts such as process management, memory management, and file systems, which are also key components of operating systems.

Course Objectives:

10. Understand classic distributed algorithms for synchronization, consistency, fault-tolerance, etc. Understand how modern distributed systems are designed and engineered.

Course Outcomes: After successful completion of the course, Student will be able to:

CO 1	Understand how to apply the knowledge to gain insight of Distributed System in solving real world problems.(BL-1)
CO 2	Identify and formulate the broader domain areas where the concept of Distributed System can be used.(BL-2)
CO 3	Develop various web applications and automate the real time problems.(BL-3)
CO 4	Enhance the concept of failure recovery in Distributed System and also develop software to recover from failure. (BL-4)
CO 5	Utilize the modern software and technical skills in order to control concurrency in 5distributed transactions. .(BL-5)

CO-POMapping

CO	PO												PSO	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	3	3	3	2	2		2				1	2	1	3
CO2	3	3	3	2	2		2				1	2	1	1
CO3	3	3	3	2	2		2				1	2		3
CO4	3	3	3	2	2		2				1	2	2	2
CO5	3	3	3	2	2		2				1	2	2	1

1-Low,2-Medium,3-High

COURSECONTENT		
MODULE– 1		**9H**
Characterization of Distributed Systems: Introduction, Examples of Distributed Systems, Resource Sharing and the Web, Challenges. **System Models:** Introduction, Architectural Models, Fundamental Models		
MODULE– 2		**10H**
Time and Global States: Introduction, Clocks Events and Process States, Synchronizing Physical Clocks, Logical Time and Logical Clocks, Global States, Distributed Debugging. **Coordination and Agreement:** Introduction, Distributed Mutual Exclusion, Elections, Multicast Communication, Consensus and Related Problems...		
MODULE– 3		**10H**
Inter Process Communication: Introduction, The API for the Internet Protocols, External Data Representation and Marshalling, Client-Server Communication, Group Communication, Case Study: IPC in UNIX. **Distributed Objects and Remote Invocation:** Introduction, Communication between Distributed Objects, Remote Procedure Call, Events and Notifications, Case Study: JAVA RMI.		
MODULE– 4		**11H**
Distributed File Systems: Introduction, File Service Architecture, Case Study 1: Sun Network File System, Case Study 2: The Andrew File System. **Name Services:** Introduction, Name Services and the Domain Name System, Directory Services, Case Study of the Global Name Services. **Distributed Shared Memory:** Introduction, Design and Implementation Issues, Sequential Consistency and IVY case study, Release Consistency, Munin Case Study, Other Consistency Models.		
MODULE– 5		**10H**
Transactions and Concurrency Control: Introduction, Transactions, Nested Transactions, Locks, Optimistic Concurrency Control, Timestamp Ordering, Comparison of Methods for Concurrency Control. **Distributed Transactions:** Introduction, Flat and Nested Distributed Transactions, Atomic Commit Protocols, Concurrency Control in Distributed Transactions, Distributed Deadlocks, Transaction Recovery		
	Total hours:	**50hours**

TEXTBOOK:

1. Distributed Systems, Concepts and Design, George Coulouris, J Dollimore and Tim
2. Kindberg, Pearson Education, 4th Edition, 2009.

REFERENCES:

1. Distributed Systems, Principles and Paradigms, Andrew S. Tanenbaum, Maarten VanSteen, 2nd Edition, PHI.
2. Distributed Systems, An Algorithm Approach, Sukumar Ghosh, Chapman & Hall/CRC, Taylor & Fransis Group, 2007.

NARAYANA ENGINEERING COLLEGE:GUDUR								
	GAME DEVELOPMENT							R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
21CS4005	3	0	0	48	3	40	60	100

Pre-requisite: Game developers need to be proficient in hardware technology and C, C#, C++, Java, Python, Lua and other programming languages to write code that crafts everything about the game.

Course Objectives:

11. to develop creativity and individuality in problem solving and performing tasks. to prepare students to work in teams. to prepare students to improve their skills and knowledge related to specific job positions individually. to enable students to do self-study.

Course Outcomes: After successful completion of the course, Student will be able to:

CO 1	understanding the basic Unity features relevant to 2D games and offer background Details about exporting and preparing 2D assets ready for importing into Unity.(BL-2)
CO 2	Applying how to create Scenes, Game Objects, Components and Assets and also how to create textures with transparency.(BL-3)
CO 3	understanding how to do developmental work that we'll need to perform to achieve high quality and highly controllable results in our 2D games.(BL-2)
CO 4	Applying How to solve these issues in flexible ways that do not involve compromising our power as a developer or the quality of our game.(BL-3)
CO 5	Analyze How to achieve pixel- perfect 2D games and also Learn How to configure my scene.(BL-4)

CO-PO Mapping

CO	PO												PSO	
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
CO1	2	2		1					3			1	2	2
CO2	2	2		1					3			2	2	
CO3	2	3		1					3			2	2	1
CO4	2	1		2	3			1	2			3	3	
CO5	2			2	1		1	2	2			3	1	1

1-Low, 2- Medium, 3- High

COURSE CONTENT		
MODULE – 1	**Third Dimension, Prototyping and Scripting Basics**	**9H**
Third Dimension: Getting to grips with 3D, Rigid body physics, Essential Unity concepts, The interface. Prototyping and Scripting Basics: first Unity project, A basic prototyping environment, introducing scripting, Understanding Translate, Testing the game so far, storing with prefabs, Using Instantiate () to spawn objects.		
MODULE – 2	**Creating The Environment, Player Characters and Further Scripting**	**10H**
Creating the Environment: Designing the game, Using the terrain editor, the terrain toolset, Creating the island—sun, sea, and sand.Player Characters and Further Scripting: Working with the Inspector, Anatomy of a character, Deconstructing the First-Person Controller object, Further scripting, Full example, Inter-script communication and Dot Syntax, Scripting for character movement.		
MODULE – 3	**Interactions, Collection, Inventory, and HUD (Headsup Display)**	**10H**
Interactions: External modeling applications, setting up the outpost model, Adding the outpost, Collisions and triggers, Ray casting, Opening the outpost. Collection, Inventory, and HUD: Creating the power cell prefab, scattering power cells, restricting outpost access, Displaying the power cell HUD.		
MODULE – 4	**Instantiation and Rigid Bodies, Particle Systems**	**10H**
Instantiation and Rigid bodies: Utilizing instantiation, Rigid bodies, Making the mini- game. Particle Systems: particle system, Creating the task, Testing and confirming.		
MODULE – 5	**Designing Menus, Animation Basics, and Building**	**9H**
Designing Menus: Interfaces and menus, Creating the menu with GUI Textures and mouse events, Creating the menu with the Unity GUI class and GUI skins Animation Basics: Game win sequence. Building: Build options, Build Settings, Player Settings, Quality Settings, Building the game.		
	Total hours:	**48 hours**

TEXTBOOK:

1. Will Goldstone, Unity 3.x Game Development Essentials Game development with C# and Javascript, Packet Publishing, Second Edition, 2011.

REFERENCES:

1. Sue Blackman, Beginning Game Development, Apress publisher 2nd Edition 2013.

Online Learning Resources: 1. https://www.udemy.com/topic/game-development/I

Professional Elective-II

NARAYANA ENGINEERING COLLEGE:GUDUR							
	ETHICAL HACKING						R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks	
	L	T	P		C	CIE	SEE	TOTAL
21CS4006	3	0	0	50	3	40	60	100

Pre-requisite: Fundamental knowledge of computer systems, networks, and their architecture. Basic understanding of cyber security principles, including threats, vulnerabilities, and risks.

Course Objectives:

- Explain the stages of ethical hacking, from planning to integration, within a structured framework.
- Evaluate how business objectives impact security policies and strategies.
- Implement social engineering, physical security, and internet reconnaissance strategies.

Course Outcomes: After successful completion of the course, Student will be able to:

CO 1	Describe each phase of the hacker framework, including planning, reconnaissance, enumeration, vulnerability analysis, exploitation, and final analysis**.** (BL-2)
CO 2	Evaluate how business objectives influence security policies and decision-making processes. (BL-5)
CO 3	Evaluate physical security vulnerabilities and their relevance in the reconnaissance phase. (BL-5)
CO 4	Understand and apply various enumeration techniques used in ethical hacking and penetration testing. (BL-2)
CO 5	Design mitigation strategies based on assessment results to bolster defense mechanisms. (BL-6)

CO-PO Mapping														
CO	**PO**												**PSO**	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	2	2												
CO2	2	2											1	1
CO3	1	2											1	2
CO4	1	2											1	2
CO5	1	2	2											
1-Low, 2- Medium, 3- High														

COURSE CONTENT		
MODULE – 1	**Hacker Framework**	**10H**
Introduction: Hacking Impacts, The Hacker Framework: Planning the test, Sound Operations, Reconnaissance, Enumeration, Vulnerability Analysis, Exploitation, Final Analysis, Deliverable, Integration Information Security Models: Computer Security, Network Security, Service Security, Application Security, Security Architecture Information Security Program: The Process of Information Security, Component Parts of Information Security Program, Risk Analysis and Ethical Hacking.		
MODULE – 2	**Business Perspective**	**10H**
Business Objectives, Security Policy, Previous Test Results, Business Challenges Planning for a Controlled Attack: Inherent Limitations, Imposed Limitations, Timing is Everything, Attack Type, Source Point, Required Knowledge, Multi-Phased Attacks, Teaming and Attack Structure, Engagement Planner, The Right Security Consultant, The Tester, Logistics, Intermediates, Law Enforcement.		
MODULE – 3	**Reconnaissance**	**10H**
Preparing for a Hack: Technical Preparation, Managing the Engagement Reconnaissance: Social Engineering, Physical Security, Internet Reconnaissance.		
MODULE – 4	**Enumeration**	**9H**
Enumeration Techniques, Soft Objective, Looking Around or Attack, Elements of Enumeration, Preparing for the Next Phase Exploitation: Intuitive Testing, Evasion, Threads and Groups, Operating Systems, Password Crackers, Root Kits, applications, Wardialing, Network, Services and Areas of Concern.		
MODULE – 5	**Deliverable**	**11H**
The Deliverable, The Document, Overall Structure, Aligning Findings, Presentation Integration: Integrating the Results, Integration Summary, Mitigation, Defense Planning, Incident Management, Security Policy, Conclusion.		
	Total hours:	**50 hours**

TEXTBOOK:

1. James S. Tiller, "The Ethical Hack: A Framework for Business Value Penetration Testing", Auerbach Publications, CRC Press.

REFERENCES:

1. EC-Council, "Ethical Hacking and Countermeasures Attack Phases", Cengage Learning
2. Michael Simpson, Kent Backman, James Corley, "Hands-On Ethical Hacking and Network Defense", Cengage Learning.

NARAYANA ENGINEERING COLLEGE:GUDUR								
	SOFTWARE ARCHITECTURE							R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
21CS4007	3	0	0	50	3	40	60	100

Pre-requisite: Knowledge on Software Engineering, Basics concepts of computer

Course Objectives:

- Introduction to the fundamentals of software architecture.
- Software architecture and quality requirements of a software system
- Fundamental principles and guidelines for software architecture design, architectural styles, patterns, and frameworks.
- Methods, techniques, and tools for describing software architecture and documenting design rationale.
- Software architecture design and evaluation processes.

Course Outcomes: After successful completion of the course, Student will be able to:

CO 1	Demonstrate Software Architecture Reference Models and Architecture Business Cycle for making a good Software Architecture
CO 2	Choose different Software Architectural Life Cycles for designing a good Architecture
CO 3	Identify Quality Attributes, Functional attributes, and different types of tactics for creating architecture.
CO 4	Develop the document of software architecture and views for creating architecture.
CO 5	Develop real time projects by combining ATAM and CBAM frameworks with quality attributes.

CO-PO Mapping

CO	PO												PSO	
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
CO1	3	2	2	2									3	2
CO2	3	3	2	2									3	2
CO3	3	1	2	2									3	2
CO4	3	2	2	2									3	2
CO5	2	3	2	2									3	2

1-Low, 2- Medium, 3- High

COURSE CONTENT		
MODULE – 1	**SOFTWARE ARCHITECTURE**	**10H**
What is software Architecture: What is Software Architecture, Other Points of View, Architectural Patterns, Reference Models, and Reference Architectures, Importance of Software Architecture, Architectural Structures and views. **Envisioning Architecture**: Architecture Business Cycle- Architectures influences, Software Processes and the Architecture Business Cycle, Making of Good Architecture.		
MODULE – 2	**DESIGNING THE ARCHITECTURE WITH STYLES**	**10H**
Designing the Architecture: Architecture in the Life Cycle, Designing the Architecture, Formatting the Team Structure, Creating a Skeletal System. **Architecture Styles:** Architectural Styles, Pipes and Filters, Data Abstraction and Object Oriented Organization, Event-BasedImplicit Invocation, Layered Systems, Repositories, Interpreters.		
MODULE – 3	**CREATING AN ARCHITECTURE-I**	**10H**
Creating an Architecture: Understanding Quality Attributes –Functionality and Architecture, Architecture and Quality Attributes, System Quality Attributes, Quality Attribute. Scenarios in Practice, Other System Quality Attributes, Business Qualities, Architecture Qualities. **Achieving Qualities**: Introducing Tactics, Availability Tactics, Modifiability Tactics, Performance Tactics, Security Tactics, Testability Tactics, Usability Tactics.		
MODULE – 4	**CREATING AN ARCHITECTURE-II**	**10H**
Documenting Software Architectures: Use of Architectural Documentation, Views, Choosing the Relevant Views, Documenting a view, Documentation across Views. **Reconstructing Software Architecture:** Introduction, Information Extraction, Database Construction, View Fusion, and Reconstruction.		
MODULE – 5	**ANALYZING ARCHITECTURES**	**10H**
The ATAM: Participants in the ATAM, Outputs of The ATAM, Phases Of the ATAM. **The CBAM:** Decision-Making Context, The Basis for the CBAM, Implementing the CBAM. **The World Wide Web:**A Case study in Interoperability- Relationship to the Architecture Business Cycle, Requirements and Qualities, Architecture Solution, Achieving Quality Goals.		
	Total Hours:	**50Hours**

TEXTBOOK:

1. Software Architectures in Practice, Len Bass, Paul Clements, Rick Kazman, 2nd Edition,
   Pearson Publication.
2. Software Architecture, Mary Shaw and David Garlan, First Edition, PHI Publication, 1996.

REFERENCES:

1.Software Design: From Programming to Architecture, Eric Braude, Wiley, 2004.

2. N. Domains of Concern in Software Architectures and Architecture Description Languages.

Medvidovic and D. S. Rosenblum. USENIX.

NARAYANAENGINEERINGCOLLEGE:GUDUR								
	BUSINESS INTELLIGENCE AND ANALYTICS							R21
Course Code	Hours/Week			Total hrs	Credit	MaxMarks		
	L	T	P		C	CIE	SEE	TOTAL
21CS4008	3	0	0	48	3	40	60	100

Pre-requisite: One of the most important prerequisites for data analytics is a strong foundation in mathematics and statistics. A solid understanding of probability theory, linear algebra, calculus, and statistical analysis is essential for working with large data sets and drawing meaningful insights**.**

CourseObjectives:

1. Gain a thorough understanding of fundamental concepts in Business Intelligence and Analytics,
2. Develop proficiency in using popular BI tools such as Tableau or Power BI for data visualization and reporting.
3. Acquire skills in applying statistical and analytical methods to analyze and interpret data.
4. Engage in hands-on case studies and real-world applications to bridge theoretical knowledge with practical scenarios.
5. Cultivate the ability to make informed decisions based on data. Understand how BI and Analytics contribute to strategic decision-making processes within an organization.

CourseOutcomes:Aftersuccessfulcompletionofthecourse,Studentwillbeableto:

CO 1	Foundational Knowledge in BI and Analytics
CO 2	Proficiency in BI Tools and Technologies
CO 3	Data Analysis and Interpretation Skills
CO 4	Problem-Solving and Decision-Making Skills
CO 5	Communication and Presentation Skills

CO-POMapping

CO	PO												PSO	
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
CO1	3	2	2	3	2	3	2	3	3	3	2	2	2	2
CO2	2		3	2	3	2	3	3	2	2	2	3	2	3
CO3	3	2	3	2	3	3	2	3	2	2	2	2	3	2
CO4	2	3	2	3	2	3	2	2	2	3	2	2	2	2
CO5	3	2	2	2	2	2	2	3	2	2	2	3	2	2
1-Low,2-Medium,3-High														

COURSECONTENT		
MODULE– 1	**Introduction to Business Intelligence**	**10H**
Understanding the scope of today's BI solutions and how they fit into existing infrastructure Assessing new options such as SaaS and cloud-based technology. Describe BI, its components & architecture, previewing the future of BI Crafting a better experience for all business users, End User Assumptions, Setting up Data for BI, The Functional Area of BI Tools, Query Tools and Reporting, OLAP and Advanced Analytics, Supporting the requirements of senior executives, including performance management.		
MODULE– 2	**Elements of Business Intelligence Solutions**	**10H**
Reports & ad hoc queries; Analyse OLAP data; Dashboards & Scorecards development, Metadata Models; Automated tasks & events; Mobile & disconnected BI; Collaboration capabilities; Real time monitoring capabilities; Software development kit; Consume BI through portals, web applications, Desktop applications.		
MODULE– 3	**Building the BI Project**	**10H**
Planning the BI project, Project Resources; Project Tasks, Risk Management and Mitigation, Cost-justifying BI solutions and measuring success, Collecting User Requirements, Requirements-Gathering Techniques; Prioritizing & Validating BI Requirements, Changing Requirements; BI Design and Development, Best Practices for BI Design; Post-Implementation Evaluations, Maintaining Your BI Environment.		
MODULE– 4	**Reporting authoring**	**(9H)**
Building reports with relational vs Multidimensional data models ; Types of Reports – List, crosstabs, Statistics, Chart, map, financial etc; Data Grouping & Sorting, Filtering Reports, Adding Calculations to Reports, Conditional formatting, Adding Summary Lines to Reports. Drill up, drill- down, drill-through capabilities. Run or schedule report, different output forms – PDF, excel, csv, xml etc.		
MODULE– 5	**BI Deployment, Administration & Security**	**9H**
Centralized Versus Decentralized Architecture, BI Architecture Alternatives, phased & incremental BI roadmap, System Sizing, Measurements and Dependencies, System Sizing, Measurements, and Dependencies. Setting Early Expectations and Measuring the Results. End-User Provisos. OLAP Implementations. Expanding BI Authentication Authorization, Access Permissions, Groups and Roles, Single-sign on Server Administration, Manage Status & Monitoring, Audit, Mail server & Portal integration, Back Up and Restore.		
	Totalhours:	**48hours**

TEXTBOOK:

1. Business Intelligence - IBM ICE Publication, 2012

REFERENCES:

1. http://en.wikipedia.org/wiki/Business_intelligence.
2. http://www.webopedia.com/TERM/B/Business_Intelligence.html.
3. Http://www.cio.com/article/40296/Business_Intelligence_Definition_and_Solutions.

NARAYANA ENGINEERING COLLEGE:GUDUR								
	SERVICE ORIENTED ARCHITECTURE							R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
20CS4009	3	0	0	48	3	40	60	100

Pre-requisite: Programming Skills

Course Objectives:

12. Understand SOA and evolution of SOA.
13. Understand web services and primitive, contemporary SOA.
14. Understand various service layers.
15. Understand service-oriented analysis and design based on guidelines.

Course Outcomes: After successful completion of the course, Student will be able to:

CO 1	Understand the fundamentals of SOA and the characteristics of Contemporary SOA(BL 2)
CO 2	Identify the framework of Web services and various activities in web services and contemporary SOA (BL 2)
CO 3	Demonstrate the interrelation between principles of service orientation (BL3).
CO 4	Examine SOA delivery Lifecycle phases and Service Modeling and Service-Oriented Design tools (BL4)
CO 5	Understand the various business service designs and WS-BPEL Language basics (BL 2)

CO-PO Mapping

CO	PO												PSO	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	3												2	2
CO2	3	2	2	2									2	2
CO3	3	3	3	2									2	2
CO4	3	3	3	3									2	2
CO5	3	2	2	2	3								2	2

1-Low, 2- Medium, 3- High

COURSE CONTENT		
MODULE – 1	**Introducing SOA**	**9H**
Introducing SOA: Fundamental SOA, Common Characteristics of Contemporary SOA, Common TangibleBenefits of SOA, Common Pitfalls of Adopting SOA. **The Evolution of SOA:** An SOA Timeline, The Continuing Evolution of SOA, The Roots of SOA.		
MODULE – 2	**Web Services and Primitive SOA**	**10H**
Web Services and Primitive SOA: The Web Services Frame Work, Services, Service Descriptions, Messaging. **Web Services and Contemporary SOA (Part I-Activity management and Composition):** Message Exchange Patterns, Service Activity, Coordination, Atomic Transactions, Orchestration, Choreography. Web Services and Contemporary SOA (Part-II-Advanced Messaging, Metadata and Security): Addressing, Reliable Messaging, Correlation, Policies, Metadata exchange, Security.		
MODULE – 3	**Principles of Service-Orientation**	**10H**
Principles of Service-Orientation: Service–Orientation and the Enterprise, Anatomy of SOA, Common Principles of Service–Orientation, Interrelation between Principles of Service-Orientation, Service Orientation and Object Orientation, Native Web Services Support for Principles of Service-Orientation. **Service Layers:** Service-Orientation and Contemporary SOA, Service Layer abstraction, Application Service Layer, Business Service Layer, Orchestration Service Layer, Agnostic Services, Service Layer ConfigurationScenarios.		
MODULE – 4	**SOA Delivery Strategies**	**(10H)**
SOA Delivery Strategies: SOA Delivery Lifecycle Phases, The Top-Down Strategy, The Bottom-up Strategy, The Agile Strategy. **Service Oriented Analysis (Part I-Introduction):** Introduction to Service Oriented Analysis, Benefits of aBusiness Centric SOA, Deriving Business Services. **Service Oriented Analysis (Part-II-Service Modeling):** Service Modeling, Service Modeling Guidelines, Classifying Service Model Logic, Contrasting Service Modeling Approaches. **Service Oriented Design (Part I-Introduction):** Introduction to Service-Oriented Design, WSDL Related XML Schema Language Basics, WSDL Language Basics, Service Interface Design Tools. **Service Oriented Design (Part II-SOA Composition Guidelines):** SOA Composing Steps, Considerations for Choosing Service Layers, Considerations for Positioning Core SOA Standards, Considerations for Choosing SOA Extensions.		
MODULE – 5	**Service Oriented Design**	**9H**
Service Oriented Design (Part III- Service Design): Service Design Overview, Entity-Centric Business Service Design, Application Service Design, Task-Centric Business Service Design, Service Design Guidelines. **Service Oriented Design (Part IV-Business Process Design):** WS-BPEL Language Basics, WS- Coordination Overview, Service Oriented Business Process Design.		
	Total hours:	**48 hours**

TEXTBOOK:

1. Service-Oriented Architecture-Concepts, Technology and Design ThomasErl, PearsonEducation
2. Understanding SOA with Web Services, Eric Newcomer, Greg Lomow, Pearson

REFERENCES:

1. The Definitive guide to SOA, Jeff Davies & others, Apress, Dreamtech.
2. Java SOA Cook book, E.Hewitt, SPD.
3. SOA in Practice, N. M. Josuttis, SPD.
4. Applied SOA, M. Rosen and others, Wiley India pvt. Ltd.
5. Java Web Services Architecture, J. Mc Govern, and others, Morgan Kaufmann Publishers, Elsevier.
6. SOA for Enterprise Applications, Shankar. K, Wiley India Edition.
7. SOA-Based Enterprise Integration, W. Roshen, TMH.
8. SOA Security, K. Rama Rao, C. Prasad, dreamtech press.

NARAYANA ENGINEERING COLLEGE:GUDUR							
	BIG DATA ANALYTICS						R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks	
	L	T	P		C	CIE	SEE	TOTAL
20CS4010	3	0	0	49	3	40	60	100

Pre-requisite: Knowledge of Object Oriented Programming and Distributed Systems

Course Objectives:

1. To understand Big Data Analytics for different systems like Hadoop.
2. To learn the design of Hadoop File System.
3. To learn how to analyze Big Data using different tools.
4. To understand the importance of Big Data in comparison with traditional databases.

Course Outcomes: After successful completion of the course, Student will be able to:

CO 1	Describe advanced **concepts of JAVA** for developing distributed programs in large scale cluster of computers **(BL-2)**
CO 2	Illustrate Distributed File Systems and Hadoop File Systems for data storage and also prepare Hadoop Architecture. **(BL-3)**
CO 3	Develop the **Map Reduce Programming** for building distributed programs on clusters of computers. **(BL - 3)**
CO 4	Describe the **Anatomy of Map Reduce jobs and different logs produced** by map reduce. **(BL - 2)**
CO 5	Analyze the Case Studies of Big Data by using Map Reduce Programming **(BL - 3)**

CO-PO Mapping

CO	PO												PSO	
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
CO1	3	1											1	2
CO2	1	3	1										2	2
CO3	1	3	2										2	3
CO4	1	2	2										2	3
CO5	2	3	1		1								3	3

1-Low, 2- Medium, 3- High

COURSE CONTENT		
MODULE – 1	**Introduction**	**9H**
Distributed programming using JAVA: Quick Recap and advanced Java Programming: Generics, Threads, Sockets, Simple client server Programming using JAVA, Difficulties in developing distributed programs for large scale clusters and introduction to cloud computing.		
MODULE – 2	**HADOOP**	**9H**
Distributed File systems leading to Hadoop file system, introduction, Using HDFS, Hadoop Architecture, Internals of Hadoop File Systems.		
MODULE – 3	**MAPREDUCE**	**10H**
Map-Reduce Programming: Developing Distributed Programs and issues, why map- reduce and conceptual understanding of Map-Reduce programming, Developing Map-Reduce programs in Java, setting up the cluster with HDFS and understanding how Map- Reduce works on HDFS, Running simple word count Map-Reduce program on the cluster, Additional examples of M-R Programming.		
MODULE – 4		**11H**
Anatomy of Map-Reduce Jobs: Understanding how Map- Reduce program works, tuning Map-Reduce jobs, Understanding different logs produced by Map-Reduce jobs and debugging the Map- Reduce jobs.		
MODULE – 5		**10H**
Case studies of Big Data analytics using Map-Reduce programming: K-Means clustering, using Big Data analytics libraries using Mahout.		
	Total hours:	**49 hours**

TEXTBOOK:

1. JAVA in a Nutshell 4th Edition.
2. Hadoop: The definitive Guide by Tom White, 3rd Edition, O'reily.

REFERENCES:

1. Hadoop in Action by Chuck Lam, Manning Publications.

Professional Elective-IV

NARAYANA ENGINEERING COLLEGE:GUDUR								
	INFORMATION AND CYBER SECURITY							R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
21CS4011	3	1	0	52	3	40	60	100

Pre-requisite: Fundamentals of Networking, Cyber Security fundamentals such as Encryption, Firewalls, Authentication etc.

Course Objectives:

- Appraise the current structure of cyber security roles across the DoD enterprise, including the roles and responsibilities of the relevant organizations.
- Evaluate the trends and patterns that will determine the future state of cyber security.

Course Outcomes: After successful completion of the course, Student will be able to:

CO 1	Analyze threats and risks within context of the cyber security architecture.
CO 2	Understand the Functionalities and mechanisms behind tools such as Proxy servers, Phishing, Keyloggers etc.
CO 3	Analyze the development of Cyber Forensics and its significance in Modern Digital Investigations.
CO 4	Apply Digital forensic methodologies specific to handheld devices.
CO 5	Evaluate decision making outcomes of cyber security scenarios.

CO-PO Mapping

CO	PO												PSO	
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
CO1	2	2											1	
CO2	2	2											1	1
CO3	1	3											1	2
CO4	1	3	1										1	2
CO5	1	2												

1-Low, 2- Medium, 3- High

COURSE CONTENT		
MODULE – 1	**Mobile and Wireless devices**	**11H**
Mobile and Wireless devices-Trend mobility-authentication service security Attacks on mobile phones-mobile phone security Implications for organizations-Organizational measurement for Handling mobile-Security policies and measures in mobile computing era. Cases.		
MODULE – 2	**Tools and Methods**	**10H**
Tools and methods used in cyber crime-Proxy servers and Anonymizers-Phishing Password cracking-Key loggers and Spy wares-Virus and worms-Trojan Horse and Backdoors Steganography-SQL Injection-Buffer overflow-Attacks on wireless network. Cases		
MODULE – 3	**Computer Forensic**	**11H**
Understanding computer forensic-Historical background of cyber forensic Forensic analysis of e-mail-Digital forensic life cycle-Network forensic-Setting up a computer forensic Laboratory-Relevance of the OSI 7 Layer model to computer Forensic-Computer forensic from compliance perspectives. Cases.		
MODULE – 4	**Digital forensic**	**10H**
Forensic of Hand –Held Devices-Understanding cell phone working characteristics-Hand-Held devices and digital forensic- Toolkits for Hand-Held device-Forensic of i-pod and digital music devices-Techno legal Challenges with evidence from hand-held Devices. Cases.		
MODULE – 5	**Cyber Security and Cyber Crimes**	**10H**
Cyber Security –Organizational implications-cost of cybercrimes and IPR issues Web threats for organizations: the evils and Perils-Social media marketing Security and privacy Implications-Protecting people privacy in the organizations Forensic best practices for organizations. Cases		
	Total hours:	**52 hours**

TEXTBOOK:

1. Nina Godbole & Sunit Belapure —Cyber Security‖, Wiley India, 2012.

REFERENCES:

1. Harish Chander, —cyber laws & IT protection‖, PHI learning pvt.ltd, 2012.
2. Dhiren R Patel, —Information security theory & practice‖, PHI learning pvt ltd,2010.
3. MS. M. K. Geetha & Ms. Swapne Raman‖Cyber Crimes and Fraud Management, ‖MACMILLAN, 2012. Pankaj Agarwal : Information Security & Cyber Laws (Acme Learning), Excel, 2013.

NARAYANA ENGINEERING COLLEGE:GUDUR								
	SOFTWARE TESTING							R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
21CS4012	3	0	0	48	3	40	60	100

Pre-requisite: A prerequisite for software testing training is having a basic understanding of software development processes. Testers need to comprehend the software development life cycle (SDLC) and the different phases involved, such as requirements gathering, design, coding, testing, and deployment.

Course Objectives:

16. The primary goals of software testing are to validate the software's functionality, enhance its performance, and improve the overall user experience.

Course Outcomes: After successful completion of the course, Student will be able to:

CO 1	List a range of different software testing techniques and strategies and be able to apply specific(automated) unit testing method to the projects.(BL-3)
CO 2	Distinguish characteristics of structural testing methods.(BL-4)
CO 3	Demonstrate the integration testing which aims to uncover interaction and compatibility problems as early as possible.(BL-3)
CO 4	Discuss about the functional and system testing methods.(BL-2)
CO 5	Demonstrate various issues for object oriented testing.(BL-3)

CO-PO Mapping

CO	PO												PSO	
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
CO1	3	2	2	2									2	
CO2	3	2	2	2	2								2	
CO3	2	2	3	2									2	
CO4	3	3	2										2	
CO5	3	2	3	2	2	2							2	

1-Low, 2- Medium, 3- High

COURSE CONTENT		
MODULE – 1		**10H**
Introduction: Purpose of Testing, Dichotomies, Model for Testing, Consequences of Bugs, Taxonomy of Bugs. **Flow graphs and Path testing:** Basics Concepts of Path Testing, Predicates, Path Predicates and Achievable Paths, Path Sensitizing, Path Instrumentation, Application of Path Testing.		
MODULE – 2		**10H**
Transaction Flow Testing: Transaction Flows, Transaction Flow Testing Techniques. Dataflow testing: Basics of Dataflow Testing, Strategies in Dataflow Testing, Application of Dataflow Testing.		
MODULE – 3		**10H**
Domain Testing: Domains and Paths, Nice & Ugly Domains, Domain testing, Domains and Interfaces Testing, Domain and Interface Testing, Domains and Testability.		
MODULE – 4		**(9H)**
Paths, Path products and Regular expressions: Path Products & Path Expression, Reduction Procedure, Applications, Regular Expressions & Flow Anomaly Detection. Logic Based Testing: Overview, Decision Tables, Path Expressions, KV Charts, Specifications.		
MODULE – 5		**9H**
State, State Graphs and Transition Testing: State Graphs, Good & Bad State Graphs, State Testing, Testability Tips. **Graph Matrices and Application:** Motivational Overview, Matrix of Graph, Relations, Power of a Matrix, Node Reduction Algorithm, Building Tools.		
	Total hours:	**48 hours**

TEXTBOOK:

1. Boris Beizer, "Software testing techniques", Dreamtech, second edition, 2002

REFERENCES:

1. Brian Marick, "The craft of software testing", Pearson Education.
2. Yogesh Singh, "Software Testing", Camebridge
3. P.C. Jorgensen, "Software Testing" 3rd edition, Aurbach Publications (Dist.bySPD).
4. N.Chauhan, "Software Testing", Oxford University Press.
5. P.Ammann&J.Offutt, "Introduction to Software Testing" , Cambridge Univ.Press.
6. Perry, "Effective methods of Software Testing", John Wiley, 2nd Edition, 1999.

NARAYANA ENGINEERING COLLEGE:GUDUR								
	DATA VIRTUALIZATION TECHNIQUES							R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
21CS4013	3	0	0	50	3	40	60	100

Pre-requisite: Programming Skills

Course Objectives:

4. Understand how server, desktop, network virtualization, and cloud computing contribute to reducing data center footprint, environmental impact, and power requirements.
5. Develop skills in virtual machine deployment, modification, and management.

Course Outcomes: After successful completion of the course, Student will be able to:

CO 1	Understand the challenges associated with data management
CO 2	Understand the networking and storage aspects of virtualized environments and be able to ensure security, performance, and integration with existing infrastructure.
CO 3	Understand access control principles and security considerations specific to virtualized environments, ensuring a comprehensive understanding of virtual machine technologies.
CO 4	Understand the resource monitoring, management, and optimization, particularly focusing on memory and CPU aspects in both physical and virtualized environments.
CO 5	Explain safeguard virtual machines, ensure high availability, and effectively manage data protection strategies in enterprise virtualized environments.

CO-PO Mapping

CO	PO												PSO	
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
CO1	3	1	3										3	3
CO2	2	2	3	3									1	1
CO3	3	3	3	2									1	2
CO4	3	1	2	3									2	3
CO5	2	1	3	1									3	2

1-Low, 2- Medium, 3- High

COURSE CONTENT		
MODULE – 1		**10H**
DATA CHALLENGES: How server, desktop, network Virtualization and cloud computing reduce data center footprint, environmental impact and power requirements by driving server consolidation; Evolution of Data : The evolution of computing infrastructures and architectures from standalone servers to rack optimized blade servers and unified computing systems (UCS).		
MODULE – 2		**10H**
ENTERPRISE-LEVEL VIRTUALIZATION: Provision, monitoring and management of a virtual datacenter and multiple enterprise-level virtual servers and virtual machines through software management interfaces; Networking and Storage in Enterprise Virtualized Environments - Connectivity to storage area and IP networks from within virtualized environments using industry standard protocols..		
MODULE – 3		**10H**
VIRTUAL MACHINES & ACCESS CONTROL: Virtual machine deployment, modification, management; monitoring and migration methodologies.		
MODULE – 4		**10H**
RESOURCE MONITORING: Physical and virtual machine memory, CPU management and abstraction techniques using a hypervisor.		
MODULE – 5		**10H**
VIRTUAL MACHINE DATA PROTECTION: Backup and recovery of virtual machines using data recovery techniques; Scalability - Scalability features within Enterprise virtualized environments using advanced management applications that enable clustering, distributed network switches for clustering, network and storage expansion; High Availability : Virtualization high availability and redundancy techniques.		
	Total hours:	**50 hours**

TEXTBOOK:

1. Mickey Iqbal, "IT Virtualization Best Practices: A Lean, Green Virtualized Data Center Approach", MC Press [ISBN: 978-1583473542] 2010.

Mike Laverick, "VMware vSphere 4 Implementation" Tata McGraw-Hill Osborne Media; 1 edition [ISBN: 978-0071664523], 2010.

REFERENCES:

1. Jason W. McCarty, Scott Lowe, Matthew K. Johnson, "VMware vSphere 4 Administration Instant Reference" Sybex; 1 edition [ISBN: 978- 0470520727], 2009.
2. Brian Perry, Chris Huss, Jeantet Fields, "VCP VMware Certified Professional on vSphere 4 Study Guide" Sybex; 1 edition [ISBN: 978-0470569610], 2009.
3. Jason Kappel, Anthony Velte, Toby Velte, "Microsoft Virtualization with Hyper-V: Manage Your Datacenter with Hyper-V, Virtual PC, Virtual Server, and Application Virtualization" McGraw-Hill Osborne [ISBN: 978- 0071614030], 2009.

NARAYANA ENGINEERING COLLEGE:GUDUR								
	CLOUD COMPUTING							R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
21CS4014	3	0	0	50	3	40	60	100

Pre-requisite:
Programming skills
Familiar with databases
Basic Knowledge in Computer Networks

Course Objectives:

17. To learn how to use Cloud Services
18. To implement Concepts of Virtualization and the Cloud delivery and Deployment Models.
19. To learn cloud computing software security objectives, design principles and development practices.

Course Outcomes: After successful completion of the course, Student will be able to:

CO 1	Analyze the Cloud computing setup with its vulnerabilities and applications using different architectures
CO 2	Identify the architecture and infrastructure of cloud computing, including cloud delivery and deployment models.
CO 3	Analyze the core issues of cloud computing such as security, privacy, and interoperability.
CO 4	Identify problems, analyze, and evaluate various cloud computing solutions.
CO 5	Analyze appropriate cloud computing solutions and recommendations according to the applications used.

CO-PO Mapping

CO	PO												PSO	
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO 11	PO 12	PSO 1	PSO2
CO1	1	2											2	2
CO2		2	2										2	
CO3		1		3									1	2
CO4		2		3										
CO5		1		2										1

1-Low, 2- Medium, 3- High

COURSE CONTENT		
MODULE – 1	**Introduction to Cloud Computing**	**10H**
Cloud Computing Insights- Distributed Computing, High Performance Computing, Utility and Enterprise Grid Computing, Cluster Computing**,** Cloud Computing fundamentals, Essential Characteristics, On Demand Self Service, Location independent resource pooling, Elastic Computing, Measured Service, Comparing cloud providers with traditional IT service providers, Vendor Lock-in, security level of third party- Security issues: Government policies.		
MODULE – 2	**Architecture**	**10H**
Cloud computing architecture, Layers of Cloud computing- IaaS, PaaS and SaaS, Cloud deployment models- Private, Public, Hybrid and Community Clouds, Advantages of Cloud Computing.		
MODULE – 3	**Virtualized Environments**	**10H**
Introduction, Characteristics of Virtualized Environments, Virtualization and Cloud Computing, Pros and Cons of Virtualization, Virtual machines and Virtualization of Clusters and Data Centres, Case studies – Xen Virtual Machine monitors – Xen API, VMware- VMware products- VMware features, Microsoft Virtual Server- Features of Microsoft Virtual Server, Open stack.		
MODULE – 4	**Cloudsim**	**10H**
Cloudsim Open source framework, Simulate VMs, memory, network, disks; Aneka – Cloud computing Framework for Enterprise Cloud applications development, Aneka Architecture, Programming models: Thread, Task and MapReduce		
MODULE – 5	**Case Studies**	**10H**
Case studies – Salesforce.com for SaaS application development, GAE- Google App Engine, Microsoft Windows Azure – public resources for VMs and Services, AWS- Amazon Web Services – public cloud registration, Services, OpenStack – Open Source Development Platform for Clouds and tools.		
	Total hours:	**50 Hours**

TEXTBOOK:

1. Rajkumar Buyya, Christian Vecchiola, S. ThammaraiSelvi, "Mastering Cloud Computing – Foundations and applications", McGraw Hill Publications,
2. Toby Velte, Anthony Velte, Robert Elsenpeter, "Cloud Computing – A Practical Approach", Mc Graw Hill, Inc, New York, NY, USA.

REFERENCES:

1. Kai Hwang, Geoffrey C Fox, Jack J. Dongarra, "Distributed and Cloud Computing, Morgan Kaufmann.
2. Cloud Computing Principles and Paradigms, John Wiley & Sons publications

NARAYANAENGINEERINGCOLLEGE:GUDUR								
	INFORMATION STORAGE AND RETRIEVAL SYSTEMS							R21
Course Code	Hours/Week			Total hrs	Credit	MaxMarks		
	L	T	P		C	CIE	SEE	TOTAL
21CS4015	3	0	0	48	3	40	60	100

Pre-requisite:ProgrammingSkills

Course Objectives: At the end of the course, the students will be able to:

1. To learn the important concepts and algorithms in IRS
2. To understand the data/file structures those are necessary to design, and implement information retrieval (IR) systems.

CourseOutcomes:Aftersuccessfulcompletionofthecourse,Studentwillbeableto:

CO 1	**Define** the Information retrieval system and its objectives along with various capabilities. [Remember]
CO 2	**Understand** to apply IR principles to locate relevant information from large collections of data using various indexing process and data structures. [Understand]
CO 3	**Implement** to design different document clustering algorithms. [Apply]
CO 4	**Analyze** different retrieval systems for web search tasks. [Analyze]
CO 5	**Investigate** various information retrieval systems which falls under Multimedia retrieval systems. [Create]

CO-POMapping

CO	PO												PSO	
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
CO1	3			3				2	2	2		3	2	
CO2		3						1		2				3
CO3		3	3			2			2	2		1	3	2
CO4	2	2	2				1	2	1					
CO5	3	3		3	1			2				1	2	3

1-Low,2-Medium,3-High

COURSECONTENT		
MODULE– 1		**10H**
Introduction to Information Retrieval Systems: Definition of Information Retrieval System, Objectives of Information Retrieval Systems, Functional Overview, Relationship to Database Management Systems, Digital Libraries and Data Warehouses Information Retrieval System Capabilities: Search Capabilities, Browse Capabilities, Miscellaneous Capabilities		
MODULE– 2		**10H**
Cataloging and Indexing: History and Objectives of Indexing, Indexing Process, Automatic Indexing, Information Extraction Data Structure: Introduction to Data Structure, Stemming Algorithms, Inverted File Structure, N-Gram Data Structures, PAT Data Structure, Signature File Structure, Hypertext and XML Data Structures, Hidden Markov Models		
MODULE– 3		**10H**
Automatic Indexing: Classes of Automatic Indexing, Statistical Indexing, Natural Language, Concept Indexing, Hypertext Linkages Document and Term Clustering: Introduction to Clustering, Thesaurus Generation, Item Clustering, Hierarchy of Clusters		
MODULE– 4		**(9H)**
User Search Techniques: Search Statements and Binding, Similarity Measures and Ranking,Relevance Feedback, Selective Dissemination of Information Search, Weighted Searches of Boolean Systems, Searching the INTERNET and Hypertext Information Visualization: Introduction to Information Visualization, Cognition and Perception, Information Visualization Technologies		
MODULE– 5		**9H**
Text Search Algorithms: Introduction to Text Search Techniques, Software Text Search Algorithms, Hardware Text Search Systems Multimedia Information Retrieval: Spoken Language Audio Retrieval, Non-Speech Audio Retrieval, Graph Retrieval, Imagery Retrieval, Video Retrieval		
	Totalhours:	**48hours**

TEXTBOOK:

1.Information Storage and Retrieval Systems – Theory and Implementation, Second Edition,
Gerald J. Kowalski, Mark T. Maybury, Springer

REFERENCES:

1.Frakes, W.B., Ricardo Baeza-Yates: Information Retrieval Data Structures and Algorithms,
Prentice Hall, 1992.
2. Information Storage & Retrieval By Robert Korfhage – John Wiley & Sons.
3. Modern Information Retrieval By Yates and Neto Pearson Education.

Professional Elective-IV

NARAYANA ENGINEERING COLLEGE: GUDUR								
	COMPUTER FORENSICS							R21
Course Code	Hours/Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
21CS4016	4	0	0	50	4	40	60	100

Pre-requisite: Cyber Security and Forensics.

Course Objectives:

1. Designing procedures at a suspected crime scene which helps you to ensure that the digital evidence obtained is not corrupted. Data acquisition and duplication: Recovering deleted files and deleted partitions from digital media to extract the evidence and validate them.

CourseOutcomes:Aftersuccessfulcompletionofthecourse,Studentwillbeableto:

CO 1	Understand the basic terminology of cybercrimes.(BL-2)
CO 2	Apply a number of different computer forensic tools to a given scenario.(BL-3)
CO 3	Understand the basics of computer forensics.(BL-2)
CO 4	Analyze and validate digital evidence data.(BL-3)
CO 5	Analyze acquisition methods for digital evidence related to system security.(BL-3)

CO-PO Mapping														
CO	**PO**												**PSO**	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1				3	3	3						3		
CO2				3	3	3						3		
CO3				3	3	3						3	3	3
CO4				3	3	3						3	3	3
CO5				3	3	3						3	3	3
1-Low,2-Medium,3-High														

COURSECONTENT		
MODULE– 1		**10H**
Computer Forensics Fundamentals: what is computer forensics, use of Computer Forensics in Law Enforcement, Computer Forensics Assistance to Human Resources/Employment Proceedings, Computer Forensics Services, Benefits of Professional Forensics Methodology, Steps taken by Computer Forensics Specialists **Types of Computer Forensics Technology**: Types of Military Computer Forensic Technology, Types of Law Enforcement – Computer Forensic Technology – Types of Business Computer Forensic Technology. **Computer Forensics Evidence and Capture:** Data Recovery Defined –Data Back-up and Recovery – The Role of Back-up in Data Recovery – The Data-Recovery Solution.		
MODULE– 2		**11H**
Evidence Collection and Data Seizure: Why Collect Evidence? Collection Options – Obstacles – Types of Evidence – The Rules of Evidence – Volatile Evidence – General Procedure – Collection and Archiving – Methods of Collection – Artifacts – Collection Steps – Controlling Contamination: The Chain of Custody **Duplication and Preservation of Digital Evidence:** Preserving the Digital Crime Scene –Computer Evidence Processing Steps – Legal Aspects of Collecting and Preserving Computer Forensic Evidence **Computer Image Verification and Authentication:** Special Needs of Evidential Authentication – Practical Consideration – Practical Implementation		
MODULE– 3		**9H**
Computer Forensics analysis and validation: Determining what data to collect and analyze, validating forensic data, addressing data-hiding techniques, performing remote acquisitions. **Network Forensics:** Network forensics overview, performing live acquisitions, developing standard procedures for network forensics, using network tools, examining the honey net project. **Processing Crime and Incident Scenes:** Identifying digital evidence, collecting evidence in private-sector incident scenes, processing law enforcement crime scenes, preparing for a search, securing a computer incident or crime scene, seizing digital evidence at the scene, storing digital evidence, obtaining a digital hash, reviewing a case		
MODULE– 4		**10H**
Current Computer Forensic tools: evaluating computer forensic tool needs, computer forensics software tools, computer forensics hardware tools, validating and testing forensics software **E-Mail Investigations:** Exploring the role of e-mail in investigation, exploring the roles of the client and server in e-mail, investigating e-mail crimes and violations, understanding e-mail servers, using specialized e-mail forensic tools **Cell phone and mobile device forensics:** Understanding mobile device forensics, understanding acquisition procedures for cell phones and mobile devices.		
MODULE– 5		**10H**

Working with Windows and DOS Systems: understanding file systems, exploring Microsoft File Structures, Examining NTFS disks, Understanding whole disk encryption, windows registry, Microsoft startup tasks, MS-DOS startup tasks, virtual machines.	
Total hours:	**50hours**

TEXTBOOK:

1. Computer Forensics, Computer Crime Investigation by John R. Vacca, Firewall Media, New Delhi.
2. Computer Forensics and Investigations by Nelson, Phillips Enfinger, Steuart, CENGAGE Learning

REFERENCES:

1. Real Digital Forensics by Keith J. Jones, Richard Bejtlich, Curtis W. Rose, Addison- Wesley Pearson Education
2. Forensic Compiling, A Tractitioneris Guide by Tony Sammes and Brian Jenkinson, Springer International edition.
3. Computer Evidence Collection & Presentation by Christopher L.T. Brown, Firewall Media.
4. Homeland Security, Techniques & Technologies by Jesus Mena, Firewall Media.
5. Software Forensics Collecting Evidence from the Scene of a Digital Crime by Robert M. Slade, TMH 2005
6. Windows Forensics by Chad Steel, Wiley India Edition.

NARAYANA ENGINEERING COLLEGE:GUDUR								
	OBJECT ORIENTED ANALYSIS AND DESIGN							R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
21CS4017	3	0	0	48	3	40	60	100

Pre-requisite: OOPS CONCEPTS

Course Objectives:

1. To understand how to solve complex problems
2. Analyze and design solutions to problems by using object oriented approach
3. To study the Unified Modeling Language and use it for problem solving

Course Outcomes: After successful completion of the course, Student will be able to:

CO 1	Find solutions to the complex problems using object oriented approach.
CO 2	Analyze and Design Solutions to Problems Using Object Oreinted Approach.
CO 3	Demonstrate the Notations of Unified Modeling Language.
CO 4	Student must be able to design state chart Diagrams, Component Diagram and Deployment Diagram.
CO 5	Design and Implement various Structural and Behavioral diagrams in real time Applications.

CO-PO Mapping

CO	PO												PSO	
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
CO1	3	2												3
CO2	1	3	3		3									2
CO3	2	3	3										3	2
CO4	2	3	1		1								3	1
CO5	1	3			1								3	2

1-Low, 2- Medium, 3- High

COURSE CONTENT		
MODULE – 1	**Introduction**:	**10H**
The Structure of Complex systems, The Inherent Complexity of Software, Attributes of Complex System, Organized and Disorganized Complexity, Bringing Order to Chaos, Designing Complex Systems, Evolution of Object Model, Foundation of Object Model, Elements of Object Model, Applying the Object Model.		
MODULE – 2	**Classes and Objects**:	**10H**
Nature of object, Relationships among objects, Nature of a Class, Relationship among Classes, Interplay of Classes and Objects, Identifying Classes and Objects, Importance of Proper Classification, Identifying Classes and Objects, Key abstractions and Mechanisms.		
MODULE – 3	**Introduction to UML**:	**10H**
Why model, Conceptual model of UML, Architecture, Classes, Relationships, Common Mechanisms, Class diagrams, Object diagrams.		
MODULE – 4	**Structural Modeling**:	**(9H)**
Package Diagram, Composite Structure Diagram, Component Diagram, Deployment Diagram, Profile Diagram.		
MODULE – 5	**Behavioral Modeling**:	**9H**
Use Case Diagram, Activity Diagrams, State Machine Diagrams, Sequence Diagram, Communication Diagram, Timing Diagram, Interaction Overview Diagram.		
	Total hours:	**48 hours**

TEXTBOOK:

1."Object- Oriented Analysis And Design with Applications", Grady BOOCH, Robert A. Maksimchuk, Michael W. ENGLE, Bobbi J. Young, Jim Conallen, Kellia Houston, PEARSON, 3rd edition, 2013.

2. "The Unified Modeling Language User Guide", Grady Booch, James Rumbaugh, Ivar Jacobson, PEARSON 12th Impression, 2012.

REFERENCES:

1.Object-oriented analysis and design using UML", Mahesh P. Matha, PHI

2. "Head first object-oriented analysis and design", Brett D. McLaughlin, Gary Pollice, Dave West, O'Reilly

3. "Object-oriented analysis and design with the Unified process", John W. Satzinger, Robert B. Jackson, Stephen D. Burd, Cengage Learning

4. "The Unified modeling language Reference manual", James Rumbaugh, Ivar Jacobson, Grady Booch, Addison-Wesley

NARAYANAENGINEERINGCOLLEGE:GUDUR								
	REINFORCEMENT LEARNING							R21
Course Code	Hours/Week			Total hrs	Credit	MaxMarks		
	L	T	P		C	CIE	SEE	TOTAL
21CS4018	3	0	0	48	3	40	60	100

Pre-requisite:ProgrammingSkills

Course Objectives:

1. Understand the Fundamentals of Reinforcement Learning
2. Implement and Evaluate RL Algorithms
3. Explore Advanced Topics in Reinforcement Learning
4. Real-World Applications and Case Studies
5. Ethical and Responsible AI

CourseOutcomes:Aftersuccessfulcompletionofthecourse,Studentwillbeableto:

CO 1	Students should demonstrate a solid understanding of fundamental concepts in reinforcement learning
CO 2	Students should be able to implement and experiment with key RL algorithms, including but not limited to Q-learning,
CO 3	Capability to apply reinforcement learning techniques to real-world problems
CO 4	Ability to critically evaluate the performance of RL algorithms using appropriate metrics and benchmarks.
CO 5	Awareness of ethical considerations in RL, including issues related to bias, fairness, transparency, and accountability.

CO-POMapping

CO	PO												PSO	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	2	3	3	3	3	2	2	2	2	3	3	2	3	2
CO2	3	2	2	2	3	2	3	2	3	3	2	2	3	3
CO3	2	2	2	3	2	2	2	3	3	2	3	3	2	2
CO4	2	2	2	2	2	3	3	3	2	2	2	3	2	2
CO5	2	3	3	3	2	2	2	3	3	3	3	2	2	2

1-Low,2-Medium,3-High

COURSECONTENT		
MODULE–1	**Introduction and Basics of RL**	**10H**
Defining RL Framework, Probability Basics: Probability Axioms, Random Variables, Probability Mass Function, Probability Density Function, Cumulative Distribution Function and Expectation. Introduction to Agents, Intelligent Agents – Problem Solving – Searching, Logical Agents.		
MODULE–2	**Markov Decision Process and Dynamic Programming**	**10H**
Markov Property, Markov Chains, Markov Reward Process (MRP), Bellman Equations for MRP, Dynamic Programming: Polices (Evaluation, Improvement, Iteration, Value Iteration), A synchronous Dynamic Programming, Generalized Policy Iteration, Efficiency of Dynamic Programming.		
MODULE–3	**Monte Carlo Methods and Temporal Difference Learnings**	**10H**
Monte Carlo: Prediction, Estimation of Action Values, Control and Control without Exploring Starts, Off-Policy Control, Temporal Difference Prediction: TD(0), SARSA: On-Policy TD control, Q-Learning: Off-Policy TD control, Games, Afterstates, and Other Special Cases		
MODULE–4	**Deep Reinforcement Learning**	**(9H)**
Deep Q-Networks, Double Deep-Q Networks Policy Optimization in RL: Introduction to Policy-based Methods, Vanilla Policy Gradient, REINFORCE Algorithm and Stochastic Policy Search, A synchronous Actor-Critic and Asynchronous Advantage Actor- Critic (A2C, A3C)		
MODULE–5	**Multi Agent in RL**	**9 H**
Multi-Agent Learning, Meta-learning, Partially Observable Markov Decision Process, Ethics in RL, Applying RL for Real-World Problems.		
	Totalhours:	**48hours**

TEXTBOOK:

1. Richard S. Sutton and Andrew G. Barto, "Reinforcement learning: An Introduction", Second Edition, MIT Press, 2019.
2. Ian Good fellow, YoshuaBengio, and Aaron Courville. "Deep learning." MIT press, 2017.
3. Marco Wiering, Martijn van Otterlo(Ed),"Reinforcement Learning, State-of-the-Art, Adaptation, Learning, and Optimization book series, ALO, volume 12, Springer, 2012.

REFERENCES:

1. Keng, Wah Loon, Graesser, Laura, "Foundations of Deep Reinforcement Learning: Theory and Practice in Python", Addison Wesley Data & Analytics Series, 2020.
2. Francois Chollet, “Deep Learning with Python”, Manning Publications, 2018.
3. Ragav Venkatesan, Baoxin Li, “Convolutional Neural Networks in Visual Computing”, CRC Press, 2018

NARAYANA ENGINEERING COLLEGE:GUDUR								
	HIGH PERFORMANCE COMPUTING							R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
21CS4019	3	1	0	50	3	40	60	100

Pre-requisite: Students should have knowledge of the following concepts to learn this subject. Operating System, Computer Organization and Architecture, Microprocessor and Architecture, Data structure and algorithms.

Course Objectives:

1. To Study various computing technology architecture.
2. To know Emerging trends in computing technology.
3. To highlight the advantage of deploying computing technology.

Course Outcomes: After successful completion of the course, Student will be able to:

CO 1	Understand High Performance Computing (HPC) system architectures and various computational models.
CO 2	Design and Develop an efficient parallel algorithm to solve given problem
CO 3	Illustrate data communication operations on various parallel architecture
CO 4	Analyze and measure performance of modern parallel computing systems
CO 5	Analyze the performance of HPC applications

CO-PO Mapping

CO	PO												PSO	
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
CO1	3	3	2										3	3
CO2	2	2	3										2	2
CO3	2	2	2										2	2
CO4	3	3											2	3
CO5	3	2											3	3

1-Low, 2- Medium, 3- High

COURSE CONTENT		
MODULE – 1		**9H**
Modern Processors : Stored Program Computer Architecture-General purpose cache- based microprocessor-Performance based metrics and benchmarks- Moore's Law- Pipelining- Super scalarity-SIMD- Memory Hierarchies Cache- mapping- prefetch- Multi core processors- Multithreaded processors- Vector Processors-Design Principles- Maximum performance estimates- Programming for vector architecture		
MODULE – 2		**9H**
Basic optimization techniques for serial code : scalar profiling function and line based runtime profiling-hardware performance counters- common sense optimizations- simple measures, large impact- elimination of common sub expressions- avoiding branches using simd instruction sets- the role of compilers – general optimization options- in lining - aliasing- computational accuracy register optimizations-using compiler logs- c++ optimizations -temporaries- dynamic memory management- loop kernels and iterators data access optimization: balance analysis and light speed estimates- storage order- case study: Jacobi algorithm and dense matrix transpose.		
MODULE – 3		**10H**
Parallel Computers : Taxonomy of parallel computing paradigms- Shared memory computers- Cache coherence- UMA –ccNUMA Distributed-memory computers- Hierarchical systems- Networks-Basic performance characteristics- Buses- Switched and fat- tree networks- Mesh networks- Hybrids - Basics of parallelization -Why parallelize - Data Parallelism - Function Parallelism- Parallel Scalability- Factors that limit parallel execution- Scalability metrics- Simple scalability laws- parallel efficiency – serial performance Vs Strong scalability- Refined performance models-Choosing the right scaling baseline- Case Study: Can slow processors compute faster- Load balance.		
MODULE – 4		**11H**
Distributed memory parallel programming with MPI: message passing - introduction to MPI – example - messages and point-to point communication - collective communication – non blocking point-to-point communication- virtual topologies – MPI parallelization of Jacobi solver- MPI implementation – performance properties ion Examples. Efficient MPI programming : MPI performance tools communication parameters-Synchronization, serialization, contention- Reducing communication overhead- optimal domain decomposition-Aggregating messages – Non blocking Asynchronous communication- Collective communication-Understanding intra-node point-to-point communication		
MODULE – 5		**11H**
Shared memory parallel programming with Open MP : introduction to Open MP - parallel execution - data scoping- Open MP work sharing for loops- synchronization - reductions - loop scheduling -tasking - case study: Open MP- parallel Jacobi algorithm- advanced open Mp wave front parallelization- Efficient Open MP programming: Profiling Open MP programs - Performance pitfalls ,Case study: Parallel Sparse matrix-vector multiply.		
Total hours:		**50 hours**

TEXTBOOK:

1. Georg Hager, Gerhard Wellein, Introduction to High Performance Computing for Scientists and Engineers, Chapman & Hall / CRC Computational Science series, 2011.

REFERENCES:

1. Charles Severance, Kevin Dowd, High Performance Computing, O'Reilly Media, 2nd Edition, 1998.
2. Kai Hwang, Faye Alaye Briggs, Computer Architecture and Parallel Processing, McGraw Hill, 1984.

NARAYANA ENGINEERING COLLEGE:GUDUR								
	AUGUMENTED AND VIRTUAL REALITY							R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
21CS4020	3	0	0	48	3	40	60	100

Pre-requisite: For VR development, you'll need a VR headset, such as the Oculus Rift or HTCVive, and a powerful computer to run the VR software. For AR development, you'll need a smartphone or tablet with a camera, and an AR development platform such as Unity or Vuforia.

Course Objectives:

20. To learn the fundamentals of sensation, perception, and perceptual training. To have the scientific, technical, and engineering aspects of augmented and virtual reality systems

Course Outcomes: After successful completion of the course, Student will be able to:

CO 1	Demonstrate the technical knowledge to identify problems in the field of Information Technology and its allied areas. (BL-2)
CO 2	Use literature to identify the objective, scope and the concept of the work. (BL-3)
CO 3	Analyze and formulate technical projects with a comprehensive and systematic approach. (BL-4)
CO 4	Identify the modern tools to implement technical projects. (BL-5)
CO 5	Design engineering solutions for solving complex engineering problems. (BL-6)

CO-PO Mapping

CO	PO												PSO	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	2	2	1	1		1			3	1	1	1	2	2
CO2	2	2	1	1		1			3	2	2	2	2	
CO3	2	3	1	1		2			3	2	2	2	2	1
CO4	2	1	3	2	3	1		1	2	2	1	3	3	
CO5	2			2	1	1	1	2	2	3	3	3	1	1

1-Low, 2- Medium, 3- High

COURSE CONTENT		
MODULE – 1		**10H**
How Humans interact with Computers: Common term definition, introduction, modalities through the ages (pre- 20th century, through world war-II, post-world war-II, the rise of personal computing, computer miniaturization), why did we just go over all of this? Types of common HCI modalities, new modalities, the current state of modalities for spatial computing devices, current controllers for immersive computing systems, a note on hand tracking and hand pose recognition. Designing for our Senses, Not our Devices: Envisioning a future, sensory technology explained, who are we building this future for?, sensory design, five sensory principles, Adobe's AR story		
MODULE – 2		**10H**
Virtual Reality for Art: A more natural way of making 3D art, VR for animation. 3D art optimization: Introduction, draw calls, using VR tools for creating 3D art, acquiring 3D models vs making them from scratch. How the computer vision that makes augmented reality possible works: Who are we?, a brief history of AR, how and why to select an AR platform, mapping, platforms, other development considerations, the AR cloud.		
MODULE – 3		**10H**
Virtual reality and augmented reality: cross platform theory: Why cross platform? The role of game engines, understanding 3D graphics, portability lessons from video game design, simplifying the controller input. Virtual reality toolkit: open source framework for the community: What is VRTK and why people use it? The history of VRTK, welcome to the steam VR unity toolkit, VRTK v4, the future of VRTK, success of VRTK. Three virtual reality and augmented reality development practices: Developing for virtual reality and augmented reality, handling locomotion, effective use of audio, common interaction paradigms		
MODULE – 4		**10H**
Data and machine learning visualization design and development in spatial computing: Introduction, understanding data visualization, principles for data and machine learning visualization design and development in spatial computing, why data and machine learning visualization works in spatial computing, 2D data visualization vs 3D data visualization in spatial computing, interactivity in data visualizations and in spatial computing, animation, failures in data visualization, good data visualization design optimize 3D spaces, how to create data visualization: data visualization creation pipeline, webXR, data visualization challenges in XR, data visualization industry use case examples of data visualization		
MODULE – 5		**10H**
Character AI and Behaviors: Introduction, behaviors, current practice: Reactive AI, more intelligence in the system, Deliberative AI, machine learning. The virtual and augmented reality health technology ecosystem: VR/AR health technology application design, standard UX isn't intuitive, tutorial: insight Parkinson's experiment, companies, case studies from leading academic institutions		
	Total hours:	**50 hours**

TEXTBOOK:

1. Erin Pangilinan, Steve lukas, and Vasanth Mohan, "Creating Augmented & Virtual Realities", 1st edition, O'REILLY, 2019.

REFERENCES:

Steve Aukstakalnis, "Practical Augmented Reality", Pearson Education, 2017.

Professional Elective-V

NARAYANA ENGINEERING COLLEGE:GUDUR							
	BLOCKCHAIN TECHNOLOGY						R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks	
	L	T	P		C	CIE	SEE	TOTAL
21CS4021	3	0	0	48	3	40	60	100

Pre-requisite: Having a solid understanding of how peer-to-peer networks use

Course Objectives:

1. The block chain technology course allows the students to explore the driving force behind the cryptocurrency Bitcoin. Along with the Decentralization, Cryptography, Bitcoins with its alternative coins, Smart contracts and outside of currencies

Course Outcomes: After successful completion of the course, Student will be able to:

CO 1	Discuss the cryptographic building blocks of block chain Technology.(BL-2)
CO 2	Explain the fundamental concepts of block chain Technology.(BL-2)
CO 3	Summarize the classification of consensus algorithms.(BL-2)
CO 4	Explain the concepts of first decentralized crypto-currency Bitcoin.(BL-2)
CO 5	Explain the use of smart contracts and its use cases.(BL-2)

CO-PO Mapping

CO	PO												PSO	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	2	2												1
CO2	2	2												1
CO3	2	2												1
CO4	2	2												1
CO5	2	2												1

1-Low, 2- Medium, 3- High

COURSE CONTENT		
MODULE – 1		**10H**
Blockchain concepts: Blockchain, Blockchain application example: Escrow, Blockchain stack, from web 2.0 to the next generation decentralized web, domain specific Blockchain application, Blockchain benefits and challenges. Blockchain application templates: Blockchain application components, design methodology for Blockchain applications, Blockchain applications templates		
MODULE – 2		**10H**
Setting up Ethereum development tools: Ethereum clients, Ethereum languages, Test RPC, Mist Ethereum walle, meta mask, web3 JavaScript API, truffle. Ethereum Accounts: Ethereum Accounts, key pairs, working with EOA Accounts, working with contract accounts.		
MODULE – 3		**10H**
Smart contracts: Smart contract, structure of a contract, setting up and interacting with a contract using Geth client, setting up and interacting with a contract using Mist Wallet		
MODULE – 4		**9H**
Smart contracts (continued): Smart contract examples, Smart contract patterns. Decentralized Applications: implementing Dapps, case studies,		
MODULE – 5		**9H**
Mining: Consensus on Blockchain network, mining, Block validation, state storage in Ethereum.		
	Total hours:	**48 hours**

TEXTBOOK:

1.Arshadeepbahga, Vijay madisetti, "Blockchain Applications A hands-on approach", VPT 2017.
2. Chandra mouli Subramanian, Asha A George, Abhilash K A and Meena Karthikeyan, "Blockchain Technology", University Press, 2021

REFERENCES:

1. Imran Bashir, "Mastering Blockchain" Packt Publishing Ltd, March 2017.
2. Melanie swan, "Blokchain blueprint for a new economy", O'REILLY

NARAYANA ENGINEERING COLLEGE:GUDUR								
	DevOps							R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
21CS4022	4	1	0	50	4	40	60	100

Pre-requisite: Programming Skills

Course Objectives:

21. Develop technical expertise in deploying, managing and monitoring cloud applications
22. Learn to review deployment methodologies, CI/CD pipelines & observability and use Devops tools like Git, Docker & Jenkins.

Course Outcomes: After successful completion of the course, Student will be able to:

CO 1	Understand various phases of software Development Life Cycle through Agile Software Development
CO 2	Understand Devops Architecture, Deployment and delivery process
CO 3	Implementation of Devops in various project Applications
CO 4	Develop CI/CD Practices in various applications
CO 5	Understand various stages of Devops Maturity Models & Assessment

CO-PO Mapping

CO	PO												PSO	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	3	2	2	1	2						3	2	3	3
CO2	3	2	3	2	2						3	2	3	3
CO3	3	1	1	1	2						3	2	3	3
CO4	3	1	1	1	2						3	2	3	3
CO5	3	2	2	1	1						3	2	3	3

1-Low, 2- Medium, 3- High

COURSE CONTENT		
MODULE – 1		**10H**
Phases of Software Development life cycle. Values and principles of agile software development.		
MODULE – 2		**10H**
Fundamentals of DevOps: Architecture, Deployments, Orchestration, Need, Instance of applications, DevOps delivery pipeline, DevOps eco system.		
MODULE – 3		**10H**
DevOps adoption in projects: Technology aspects, Agiling capabilities, Tool stack implementation, People aspect, processes		
MODULE – 4		**10H**
CI/CD: Introduction to Continuous Integration, Continuous Delivery and Deployment , Benefits of CI/CD, Metrics to track CICD practices		
MODULE – 5		**10H**
. Devops Maturity Model: Key factors of DevOps maturity model, stages of Devops maturity model, DevOps maturity Assessment		
	Total hours:	**50 hours**

TEXTBOOK:

1) The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations, Gene Kim , John Willis , Patrick Debois , Jez Humb,1st Edition, O'Reilly publications, 2016.
2) What is Devops? Infrastructure as code, 1st Edition, Mike Loukides, O'Reilly publications, 2012.

REFERENCES:

1) Building a DevOps Culture, 1st Edition, Mandi Walls, O'Reilly publications, 2013.
2) The DevOps 2.0 Toolkit: Automating the Continuous Deployment Pipeline With Containerized Micro services, 1st Edition, Viktor Farcic, Create Space Independent Publishing Platform publications, 2016
3) Continuous Delivery: Reliable Software Releases Through Build, Test, and Deployment Automation, 1st Edition, Jez Humble and David Farley, 2010.
4) Achieving DevOps: A Novel About Delivering the Best of Agile, DevOps, and micro services, 1st Edition, Dave Harrison, Knox Lively, A press publications, 2019

NARAYANA ENGINEERING COLLEGE::GUDUR								
	TOOLS & TECHNIQUES FOR DATA SCIENCE							R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
21CS4023	3	0	2	48	4	40	60	100

Pre requisites:

1. Knowledge in Data science
2. Knowledge in python

Course Objectives:

To understand the various concepts in Data Science process.
To study the applications of Data Science.
To learn to setup the data science tools environment and implement in Python and R
To learn to write programs in Python and R for data science projects.

Course Outcomes: After successful completion of the course, the student will be able to:

CO1**:**Understanding the data science concepts and tools used for data analysis[BL-2]

CO2: Implementation of data science Forecasting methods in supply chain management[BL-6]

CO3: Analyze the Data science concepts in education[BL-4]

CO4: Apply the Data science tools and techniques in Health care.[BL-3]

CO5: Evaluate case studies in optimization[BL-5]

COURSE CONTENT

MODULE – 1	**INTRODUCTION**	**9H**
Data Science Applications in various domains, Challenges and opportunities, tools for data scientists, Recommender systems – Introduction, methods, application, challenges.		
MODULE – 2	Forecasting in supply chain Management	**10H**
Time series data – stock market index movement forecasting. Supply Chain Management – Real world case study in logistics.		
MODULE – 3	Data science in Education	**10H**
Data Science in Education, Social media.		
MODULE – 4	Data science in Health	**9H**
Data Science in Healthcare, Bioinformatics.		
MODULE – 5	Data science optimization	**10H**
Case studies in data optimization using Python.		
	Total hours:	**48 hours**

Text Books:

1. Aakanksha Sharaff, G. K. Sinha, "Data Science and its applications ", CRC Press, 2021.
2. Q. A. Menon, S. A. Khoja, "Data Science: Theory, Analysis and Applications", CRC Press, 2020.
3. R-Programming for Data Science by Roger D. Peng (Reference).
4. The Art of R-Programming by Noman Matloff cangage learning India.
5. Python Data Science Handbook. Essential Tools for working with Data

Reference Books

1. J. Janssens, Data science at the command line, First edition. Sebastopol, CA: O'Reilly, 2014
2. C. O'Neil and R. Schutt, Doing Data Science: Straight Talk from the Frontline, 1 edition. Beijing ;

Sebastopol: O'Reilly Media, 2013.

3. J. VanderPlas, Python Data Science Handbook: Essential Tools for Working with Data, First edition. Shroff/O'Reilly, 2016.

NARAYANA ENGINEERING COLLEGE:GUDUR							
	CLOUD SECURITY						R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks	
	L	T	P		C	CIE	SEE	TOTAL
21CS4024	3	0	0	48	3	40	60	100

Pre-requisite: Programming Skills, Familiarity with Databases, Basics of Security and Privacy, Knowledge of Agile Development, Understanding of Virtualization, Basics of Networking, Basic Understanding of Different Types of Cloud

Course Objectives: As development and software delivery move rapidly toward cloud infrastructure, you must be equipped to address the challenges of security and compliance. In this course, you'll learn common cloud terminology and how to navigate the vast array of security controls you need to consider when moving to a cloud provider. By course end, you'll understand how to address common security challenges of running software in cloud infrastructure.

23.

Course Outcomes: After successful completion of the course, Student will be able to:

CO 1	Understand the various tools and methods used in cybercrime.
CO 2	Identify risk management processes, risk treatment methods, organization of information security.
CO 3	Classify cyber security solutions and information assurance.
CO 4	Examine software vulnerabilities and security solutions to reduce the risk of exploitation.
CO 5	Analyze the cyber security needs of an organization.

CO-PO Mapping

CO	PO												PSO	
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
CO1	2	2	1	1	3	3	1	3	-	1	-	3	-	3
CO2	2	2	1	1	-	3	2	3	-	1	-	3	-	3
CO3	2	2	1	1	-	3	1	3	-	1	-	3	-	3
CO4	2	2	1	1	3	3	1	3	-	1	-	3	-	3
CO5	2	2	1	1	-	3	1	3	-	1	-	3	-	3

1-Low, 2- Medium, 3- High

COURSE CONTENT		
MODULE – 1	**Cloud Computing Architectural Framework**	**9H**
Cloud Computing Architectural Framework: Cloud Benefits, Business scenarios, Cloud Computing Evolution, cloud vocabulary, Essential Characteristics of Cloud Computing, Cloud deployment models, Cloud Service Models, Multi- Tenancy, Approaches to create a barrier between the Tenants, cloud computing vendors, Cloud Computing threats, Cloud Reference Model, The Cloud Cube Model, Security for Cloud Computing, How Security Gets Integrated.		
MODULE – 2	**Cloud software security fundamentals**	**10H**
Cloud software security fundamentals: – Security objective, security service, Cloud security design principles, Secure cloud software requirements, Secure development practice, Approaches of cloud software requirements engineering, Security policy implementation, Secure cloud software testing, penetration testing, Disaster recovery, Cloud for BCP/DCP.		
MODULE – 3	**Security and Recovery**	**9H**
Traditional Security, Business Continuity, Disaster Recovery, Risk of insider abuse, Security baseline, Customers actions, Contract, Documentation, Recovery Time Objectives (RTOs), Customers responsibility, Vendor Security Process (VSP).		
MODULE – 4	**Cloud Risk Issues and Challenges**	**10H**
CIA triad, Privacy and Compliance Risk, PCIDSS, Information privacy and privacy law, Common threats and vulnerabilities, Access control issues, service provider Risk. Security policy Implementation, Computer Security incident response team (CSIRT), Virtualization security Management- virtual threats, VM security recommendations, VM security techniques – hardening, securing VM remote access.		
MODULE – 5	**Cloud Security Architecture**	**10H**
General issues, Trusted cloud, Secure execution environments and communications, Micro architecture, Identity management, Access control, Autonomic security, protection, self-healing. Cloud life cycle issues – cloud standards, DMTF, ISO, ETSI, OASI, SNIA, OGF, OWASP, Incident response, Internet Engineering Task Force Incident- Handling Guidelines, Computer security and response team, Encryption and key management, VM Architecture, Key Protection, Hardware protection, VM life cycle.		
	Total hours:	**48 hours**

TEXTBOOK:

1. Ronald L. Krutz, Russell Dean Vines, "Cloud Security", Wiley publication 2010.
2. J.R. ("Vic") Winkler, "Securing the Cloud" Syngress, 2011.
3. Tim Mather, Subra Kumaraswamy, Shahed Latif, "Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance" O'Reilly Media; 1 edition, 2009.

REFERENCES:

1. Toby Velte, Anthony Velte, Robert Elsenpeter, Cloud Computing, A Practical Approach, Tata McGraw-Hill Education, 2009.
2. GautamShroff, Enterprise Cloud Computing Technology Architecture Applications, Cambridge University Press, 2010.

NARAYANA ENGINEERING COLLEGE:GUDUR								
	VIRTUALIZATION TECHNOLOGIES							R21
Course Code	Hours / Week			Total hrs	Credit	Max Marks		
	L	T	P		C	CIE	SEE	TOTAL
21CS4025	3	0	0	50	3	40	60	100

Pre-requisite: Operating Systems

Course Objectives:

1. to introduce students to the business and technical issues associated with the formulation and implementation of security policies.

Course Outcomes: After successful completion of the course, Student will be able to:

CO 1	Apply the concept of virtualization and understand the importance of virtualization in distributed computing and how this has enabled the development of cloud computing. [BTL3]
CO 2	Manage ESX and ESXi hosts and their virtual machines using the capabilities of VMware vCenter Server. [BTL2]
CO 3	Understanding Virtual machines and Implementation of virtual machines. [BTL2]
CO 4	Understanding virtualization and various ways of using virtualization. [BTL2]
CO 5	Implementation of private cloud platform using virtualization. [BTL3]

CO-PO Mapping

CO	PO												PSO	
	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1				3	3	3						3		
CO2				3	3	3						3		
CO3				3	3	3						3	3	3
CO4				3	3	3						3	3	3
CO5				3	3	3						3	3	3

1-Low, 2- Medium, 3- High

COURSE CONTENT		
MODULE – 1		**10H**
Introduction To Virtualization System Architectures - Virtual Machine Basics- Process Virtual Machines - System Virtual Machines - Taxonomy of Virtual Machines - Emulation: Basic Interpretation – Threaded Interpretation - Pre-Coded & Direct Interpretation - Binary Translation - Full and Para-Virtualization - Types of Hypervisor - Types of Virtualization.		
MODULE – 2		**10H**
Server Virtualization Server Virtualization - Partitioning Techniques-Hardware Virtualization - Virtual Hardware -Types of Server Virtualization -Business Cases for Sever Virtualization-Uses of Virtual Server Consolidation -Selecting Server Virtualization Platform.		
MODULE – 3		**10H**
Network Virtualization Design of Scalable Enterprise Networks-Virtualizing the Campus - WAN Design-WAN Architecture - WAN virtualization -Virtual Enterprise Transport Virtualization - VLANs and Scalability - Theory Network Device Virtualization Layer 2 - VLANs Layer 3 VRF Instances Layer 2 - VFls Virtual Firewall Contexts Network Device Virtualization -Datapath Virtualization Layer 2: 802.1q-Trunking Generic Routing Encapsulation -IPSec L2TPv3Label Switched Paths-Control-Plane Virtualization -Routing Protocols -VRF- Aware Routing - Multi-Topology Routing.		
MODULE – 4		**10H**
Storage Virtualization Devices - SCSI -SCSI Communication -Using SCSI Buses - Fiber Channel -Fiber Channel Cables -Fiber Channel Hardware Devices - iSCSI Architecture – Securing iSCSI SAN Backup & Recovery Techniques - RAID -Classic Storage Model - SNIA Shared Storage Model Host based Architecture - Storage based architecture - Network based Architecture - Fault tolerance to SAN- Performing Backups - Virtual Tape Libraries		
MODULE – 5		**10H**
Applying Virtualization Comparison of Virtualization Technologies: Guest OS, Host OS, Hypervisor, Emulation, Kernel Level -Shared Kernel-Enterprise Solutions: Vmware Server, ESXi, Citrix Xen Server, Microsoft Virtual PC, Microsoft Hyper-V, Virtual Box - Server Virtualization: Configuring Server with Server Virtualization, Adjusting & Tuning Virtual Servers, VM Backup and Migration -Desktop Virtualization: Terminal Services, Hosted Desktop, Web Based Solutions, Localized Virtualized Desktop-Network and Storage Virtualization: VPN,VLAN, SAN and VSAN, NAS.		
	Total hours:	**50 hours**

TEXTBOOK:

1. Chris Wolf, Erick M. Halter, “Virtualization: From the Desktop to the Enterprise’, A Press, 2005.
2. James E. Smith, Ravi Nair, “Virtual Machines: Versatile Platforms for Systems and Processes’, Elsevier/Morgan Kaufmann, 2005.
3. David Marshall, Wade A. Reynolds, “Advanced Server Virtualization: VMware and Microsoft Platform in the Virtual Data Center’, Auerbach Publications, 2006.

REFERENCES:

1. William von Hagen, “Professional Xen Virtualization’, Wrox Publications, January, 2008.
2. Kumar Reddy, Victor Moreno, “Network virtualization”, Cisco Press, July, 2006.
3. Amy Newman, Kenneth Hess, “Practical Virtualization Solutions: Virtualization from the Trenches", Prentice Hall, October 2009.