Janardan Bhagat Shikshan Prasarak Sanstha's

CHANGU KANA THAKUR ARTS, COMMERCE & SCIENCE COLLEGE, NEW PANVEL (AUTONOMOUS)

Re-accredited 'A+' Grade by NAAC
'College with Potential for Excellence' Status Awarded by UGC
'Best College Award' by University of Mumbai

Program: Bachelor's in Science (B. Sc.)

Information Technology

Credits: 132

SYLLABUS

(Approved in the Academic council meeting held on--------)

F. Y. B. Sc. Information Technology

Revised as per
Choice Based Credit System (60:40)
w. e. f. Academic Year 2022-23

BACHELOR'S IN SCIENCE (B. Sc.)

Programme Outcomes

S. N.	After completion of B.Sc. program students will acquire	Graduate Attribute
PO1	The knowledge of the disciplines and in-depth and extensive knowledge, understanding and skills in a specific field of interest.	Disciplinary knowledge
PO2	An ability to develop and conduct experiments, analyze, and interpret data and use scientific judgment to draw conclusions	Scientific reasoning
PO3	An ability to use current technology, and modern tools necessary for creation, analysis, dissemination of information.	Digital literacy
PO4	Innovative, professional, and entrepreneurial skills needed in various disciplines of science.	Life-long learning
PO5	An ability to achieve high order communication skills.	Communication skills
PO6	An ability to collect, analyze and evaluate information and ideas and apply them in problem solving using conventional as well as modern approaches	Problem solving
PO7	A sense of social responsibility; intellectual and practical skills and demonstration of ability to apply it in real-world settings.	Reflective thinking
PO8	An ability to engage in independent and life-long learning through openness, curiosity, and a desire to meet new challenges.	Life-long learning
PO9	A capacity to relate, collaborate, and lead others, and to exchange views and ideas to work in a team to achieve desired outcomes	Teamwork
PO10	An ability to function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.	Leadership
PO11	An ability to understanding values, ethics, and morality in a multidisciplinary context.	Moral and ethical awareness

Program Specific outcomes

Name of the Programme: **B.Sc.I.T.**	**Programme Coordinator:** **Dr. J. S. Thakur**	**Head of the Department:** **Mrs. I. S. Thakare**
	After completing the programme in Information Technology, Student will be able to:	
PSO1	**Gain proficiency in the field of Networking and Security.**	
PSO2	**Develop Programming skills that help to meet the needs of the IT industry.**	
PSO3	**Build soft skills for employability and personality development in the Industrial environment.**	

Preamble:

The B.Sc. Information Technology programme is started with an aim to make the learners employable and impart industry oriented training.

The main objectives of the course are:

- To think analytically, creatively and critically in developing robust, extensible and highly maintainable technological solutions to simple and complex problems.
- To apply their knowledge and skills to be employed and excel in IT professional careers and/or to continue their education in IT and/or related post graduate programmes.
- To be capable of managing complex IT projects with consideration of the human, financial and environmental factors.
- To work effectively as a part of a team to achieve a common stated goal.
- To adhere to the highest standards of ethics, including relevant industry and organizational codes of conduct.
- To communicate effectively with a range of audiences both technical and non-technical.
- To develop an aptitude to engage in continuing professional development.

Semester - I

[Under CBCS Scheme]

Course	Course Type	Course code	Hrs. / week	Internal assessment	Semester-end examination	Total	Credits
Introduction to C++ Programming	Core Subject	UIT1CPT	5	40	60	100	2
Digital Electronics	Core Subject	UIT1DET	5	40	60	100	2
Operating Systems	Core Subject	UIT1OST	5	40	60	100	2
Discrete Mathematics	Core Subject	UIT1DMT	5	40	60	100	2
Communication Skill	Ability Enhancement Skill Course	UIT1CST	5	40	60	100	2
Course on Environmental Studies	Generic	USC1EVS	-	-	-	-	2
Introduction to C++ Programming Practical	Core Subject Practical	UIT1CPP	5	--	50	50	2
Digital Electronics Practical	Core Subject Practical	UIT1DEP	5	--	50	50	2
Operating Systems Practical	Core Subject Practical	UIT1OSP	5	--	50	50	2
Discrete Mathematics Practical	Core Subject Practical	UIT1DMP	5	--	50	50	2
Communication skill Practical	Ability Enhancement Skill Course	UIT1PCP	5	--	50	50	2

Semester - II
[Under CBCS Scheme]

Course	Course Type	Course code	Hrs/ week	Internal assessment	Semester -end examination	Total	Credits
Object Oriented Programming	Core Subject	UIT2OPT	5	40	60	100	2
Microprocessor Architecture	Core Subject	UIT2MAT	5	40	60	100	2
Database Management System	Core Subject	UIT2DST	5	40	60	100	2
Numerical Methods	Core Subject	UIT2NMT	5	40	60	100	2
Web Programming	Ability Enhancement Skill Course Practical	UIT2WPT	5	40	60	100	2
Object Oriented Programming Practical	Core Subject Practical	UIT2OPP	5	40	60	100	2
Microprocessor Architecture Practical	Core Subject Practical	UIT2MAP	5	--	50	50	2
Database Management System Practical	Core Subject Practical	UIT2DSP	5	--	50	50	2
Numerical Methods Practical	Core Subject Practical	UIT2NMP	5	--	50	50	2
Web Programming Practical	Ability Enhancement Skill Course Practical	UIT2WPP	5	--	50	50	2
Spoken English	Generic	USC2CSK	-	-	-	-	2

Examination Scheme

Choice Based Credit System (CBCS)

- **Revised Scheme of Examination**

The performance of the learners shall be evaluated into two components. The learner's Performance shall be assessed by Internal Assessment with 40% marks in the first component by conducting the Semester End Examinations with 60% marks in the second component. The allocation of marks for the Internal Assessment and Semester End Examinations are as shown below:-

A) Internal Assessment: 40 % 40 Marks

Sr. No.	Particular	Marks
01	One periodical class test examination to be conducted in the given semester	20 Marks
02	Any two tools out of these (10 Marks each) 1. Presentation and write up on the selected topics of the subjects / Case studies. 2. Quiz	20 Marks

- Maximum Marks: 20
- Duration: 30 Minutes

Particular	Marks
Match the Column / Fill in the Blanks / Multiple Choice Questions/ True/False/Answer in One or Two Lines (Concept based Questions) (1 Marks each)	20 Marks

Question Paper Pattern for Semester End Examination

(Periodical Class Test/ online examination for the Courses at Under Graduate Programmes)

- **Undergraduate Programmes for B.Sc. in Information Technology**
  - Duration: The examination shall be of 2 hours duration.

Question Paper Pattern

Theory question paper pattern
1. There shall be five questions each of 12 marks. 2. All questions shall be compulsory with internal options. 3. Question may be subdivided into sub-questions a, b, c… and the allocation of marks depends on the weightage of the unit.

- **Passing Standard**

The learners to pass a course shall have to obtain a minimum of 40% marks in aggregate for each course where the course consists of Internal Assessment and Semester End Examination. The learners shall obtain minimum of 40% marks (i.e. 16 out of 40) in the Internal Assessment and 40% marks in Semester End Examination (i.e. 24 Out of 60) separately, to pass the course and minimum of Grade D, wherever applicable, to pass a particular semester. A learner will be said to have passed the course if the learner passes the Internal Assessment and Semester End Examination together.

Question Paper Pattern for Continuous Assessment

Presentation and write-up	**Quiz**
Presentation skill	**Quiz on application of subject in real life**
Knowledge	
Quality of ppt.	
Writing skill	

Question Paper Pattern for Practical Examination

Sr. No.	Particular		Marks
01	Practical		50 Marks
	Practical Question	40 Marks	
	Journal	5 Marks	
	Viva	5 Marks	

Course Description: B.Sc. (Information Technology)	
Semester	**I**
Course Name	**Introduction to C++ programming**
Course Code	**UIT1CPT**
Eligibility for the Course	
Credit	**2**
Hours	**5Hrs per week**

Course Objectives	The objective of this course is to introduce the concept of the basic programming language with C++.

Course Outcomes	**After completing the course, Student will be able to:**
	1) Define basic concepts of C++ programming language.
	2) Illustrate different types of operators of C++ language.
	3) Explain functions in C++.
	4) Elaborate Arrays and Structures in C++.

Module/ Unit	**Course Description**	**Hrs.**	**CO No.**	**PSO No.**	**PO No.**
I	**Programming Logic and techniques** : Algorithms, Flow-charts, Program Design **Introduction to C++**: Origin of C++ C++ Program Structure, A Sample C++ program, Applications of C++, Variables and Assignments: variables, identifiers, variable declarations, local and global variables, Assignment Statements, reference variable, symbolic constant. **Input and Output:** cin, cout, escape sequences, include directives and Namespaces, Indenting and Comments, Data types, Expressions, Type compatibilities. **Operators:** Arithmetic Operators, Relational Operators, Logical Operators, Assignment Operators, Operator Precedence.	12hrs	1	3	2

II	**Manipulators:** endl, setw, sizeof, Increment and decrement operators, Type Cast Operators, Scope resolution operators. **Flow of Control:** Compound statements, Loops: while, for, do while, nested loops, Decision making: if, if – else, nested if else, switch, break and continue.	12hrs	1		
III	**Functions:** Function Definition, Function Declaration, Function Prototypes, built in functions and user defined functions, Call by reference, Call by value, const member functions. Inline Functions and recursive functions, Maths Library Functions.	12hrs	2		
IV	**Derived Data Types:** **Arrays:** Declaring Arrays, Initializing Arrays, Types of Arrays, Arrays in functions. **Pointers:** Pointers, use of pointers, Void Pointers, Null Pointers, Pointer to pointer, Passing Pointers to Functions, constant pointer, Generic Pointer.	12hrs	3		
V	**Strings and Vectors:** Strings, String functions: strcmp, strcat, strlen, strcpy. Vector Basics. **Introduction to Structures: Declaring the structure,** Structure Variables, Initialization, Structure Assignment, Nested Structure. Structures and Functions, Structures and Arrays: Arrays of Structures, Structures Containing Arrays, Unions.	12hrs	4		

References

1. "Let us C++" , Y.P.Kanetkar, Seventh edition, BPB publication
2. "Problem Solving with C++" , Walter Savitch, Sixth Edition, Pearson Education.
3. Schaum's outlines "Programming with C++", J.R.Hubbard, Second Edition, Tata McGrawHill
4. Object Oriented programming with C++ , E Balagurusamy , Third Edition ,Tata McGraw Hill.
5. Pure C++ programming , Amir Afzal, Pearson Education.
6. Computer Science – A structured Approach using C++ by B. Forouzan, R. F. Gilberg, Cengage Publication

Course Description BSc(Information Technology)	
Semester	**I**
Course Name	**Digital Electronics**
Course Code	**UIT1DET**
Eligibility for the Course	
Credit	**2**
Hours	**5Hrs per week**

Course Objectives	The objective of this course is to acquire the basic knowledge of digital logic levels and the application of knowledge to understand digital electronics circuits. To prepare the learners to perform the analysis and design of various digital electronic circuits.

Course Outcomes	**After completing the course, Student will be able to:**
	1) Define the various types of number systems & conversions.
	2) Explain the various types of logic gates along with the truth tables.
	3) Distinguish combinational and sequential logic circuits.
	4) Classify different types of flip-flops, registers and counters.

Module/ Unit	Course Description	Hrs	CO No.	PSO No.	PO No.
I	**Number System:** Analog System, digital system, numbering system, binary number system, octal number system, hexadecimal number system, conversion from one number system to another, floating point numbers, weighted codes binary coded decimal, non-weighted codes Excess – 3 code, Gray code, Alphanumeric codes –ASCII Code, EBCDIC, ISCII Code, Hollerith Code, Morse Code, Teletypewriter (TTY), Error detection and correction, Universal Product Code, Code conversion. **Binary Arithmetic and Arithmetic Circuits:** Binary addition, Binary subtraction, Negative number representation, Subtraction using 1's complement and 2's complement,Binary multiplication and division, Arithmetic in octal number system, Arithmetic in hexadecimal number system, BCD and Excess-3 arithmetic,Introduction to Arithmetic Circuits : Adder, BCD Adder, Excess – 3 Adder, Binary Subtractors, BCD Subtractor, Multiplier, Comparator	12hrs	1	1	6
II	**Boolean Algebra and Logic Gates:** Introduction, Logic (AND OR NOT), Boolean theorems, Boolean Laws, De Morgan's Theorem, Perfect Induction, Reduction of Logic expression using Boolean Algebra, Deriving Boolean expression from given circuit, exclusive OR and Exclusive NOR gates, Universal Logic gates, Implementation of other gates using universal gates, Input bubbled logic, Assertion level. **Minterm, Maxterm and Karnaugh Maps:** Introduction, minterms and sum of minterm form, maxterm and Product of maxterm form, Reduction technique using Karnaugh maps – 2/3/4/5/6 variable K-maps, Grouping of variables in K-maps, K-maps for product of sum form, minimize Boolean expression using K-	12hrs	2		

	map and obtain K-map from Boolean expression, Quine Mc Cluskey Method				
III	**Combinational Logic Circuits:** Introduction, Multi-input, multi-output Combinational circuits, Code converters design and implementations **Read only Memory :** Diode Rom, Programmable ROMs, Erasable PROMs, Programmable array logic, Programmable Logic arrays	12hrs	3		
IV	**Multiplexer, Demultiplexer, ALU, Encoder and Decoder:** Introduction, Multiplexer, Demultiplexer, Decoder, ALU, Encoders. **Sequential Circuits:** Flip-Flop: Introduction, Terminologies used, S-R flip-flop, D flip-fop, JK flipflop, Race-around condition, Master – slave JK flip-flop, T flip-flop, 12 14 | Page conversion from one type of flip-flop to another, Application of flipflops **Clocks And Timing Circuits :** Introduction to clocks : Astable, Monostable, Bistable, 555 Timer	12hrs	3		
V	**Counters:** Introduction, Asynchronous counter, Terms related to counters, IC 7493 (4-bit binary counter), Synchronous counter, Bushing, Type T Design, Type JK Design, Presettable counter, IC 7490, IC 7492, Synchronous counter ICs, Analysis of counter circuits. **Shift Register:** Introduction, parallel and shift registers, serial shifting, serial–in serial– out, serial–in parallel–out , parallel–in parallel–out, Ring counter, Johnson counter, Applications of shift registers, Pseudo-random binary sequence generator, IC7495, Seven Segment displays, analysis of shift counters.	12hrs	4		

References:

1. Digital Electronics and Logic Design, N. G. Palan, Technova
2. Make Electronics, Charles Platt, O'Reilly,1st, 2010
3. Modern Digital Electronics, R. P. Jain, Tata McGraw Hill, 3rd
4. Digital Principles and Applications, Malvino and Leach, Tata McGraw Hill
5. Digital Electronics: Principles, Devices and Applications, Anil K. Maini, Wiley, 2007

Course Description: BSc(Information Technology)	
Semester	**I**
Course Name	**Operating Systems**
Course Code	**UIT1OST**
Eligibility for the Course	
Credit	**2**
Hours	**5Hrs per week**

Course Objectives	To learn the fundamentals of Operating Systems, its functions and services. To learn the mechanisms of OS to handle processes and threads and their communication .To learn the mechanisms involved in memory management in contemporary OS.

Course Outcomes	**After completing the course, Student will be able to:**
	1) Explain the importance of Computer Systems, Process Management Policies.
	2) Define the memory management and its allocation policies.
	3) Elaborate the Cloud concepts, file systems, its structure and operations.
	4) Determine the requirement for process synchronization and coordination handled by the operating system.

Module/ Unit	Course Description	Hrs.	CO No.	PSO No.	PO No.
I	**Introduction:** What is an operating system? History of operating system, computer hardware, different operating systems, operating system concepts, system calls, operating system structure. Examples of OS: Android OS, Linux OS, Windows OS. **Processes and Threads:** Processes, threads, interprocess communication, scheduling, IPC problems.	12hrs	1	2	4
II	**Memory Management:** No memory abstraction, memory abstraction: address spaces, virtual memory, page replacement algorithms, design issues for paging systems, implementation issues, and segmentation.	12hrs	2		

III	**File Systems:** Files, directories, file system implementation, file-system management and optimization, MS-DOS file system, UNIX V7 file system, CD ROM file system. **Protection of File Systems:** Types of Access, Access Control, Other Protection Approaches.	12hrs	3		
IV	**Input-Output:** Principles of I/O hardware, Principles of I/O software, I/O software layers, disks, clocks, user interfaces: keyboard, mouse, monitor, thin clients, power management. **Deadlocks:** Resources, introduction to deadlocks, the ostrich algorithm, deadlock detection and recovery, deadlock avoidance, deadlock prevention, issues.	12hrs	4		
V	**Virtualization:** History, requirements for virtualization, type 1 and 2 hypervisors, techniques for efficient virtualization, hypervisor microkernel, memory virtualization, I/O virtualization, Virtual appliances, virtual machines on multicore CPUs **Cloud:** Introduction, Characteristics, Types of Clouds, Examples.	12hrs	1		

References:

1. Modern Operating Systems, Andrew S. Tanenbaum and Herbert Bos, 4th Edition, Pearson Publishers
2. Operating System Concepts, Abraham Silberschatz and Peter B. Galvineg Gagne, 8th Edition, Wiley Publishers
3. Operating Systems – Internals and Design Principles, Willaim Stallings, 8th Edition, Pearson Publishers
4. Operating Systems, Godbole and Kahate, 3rd Edition, McGraw Hill Publishers

Course Description: BSc(Information Technology)	
Semester	**I**
Course Name	**Discrete Mathematics**
Course Code	**UIT1DMT**
Eligibility for the Course	
Credit	**2**
Hours	**5Hrs per week**

Course Objectives	The purpose of the course is to familiarise the prospective learners with mathematical structures that are fundamentally discrete. This course introduces sets and functions, forming and solving recurrence relations and different counting principles. These concepts are useful to study or describe objects or problems in computer algorithms and programming languages.

Course Outcomes	**After completing the course, Student will be able to:**
	1) Recall basic set theory, logic, functions and relations.
	2) Solve problems using recurrence relations, counting principles and probability.
	3) Examine the properties of graphs, applications of graphs and trees.
	4) Solve the problems by using different methods of proofs, divisibility.

Module/ Unit	**Course Description**	Hrs	CO No.	PSO No.	PO No.
I	**Set Theory**: Definitions and the Element Method of Proof, Properties of Sets, Disproofs, Algebraic Proofs, Boolean Algebras, Russell's Paradox and the Halting Problem. **The Logic of Compound Statements**: Logical Form and Logical Equivalence, Conditional Statements, Valid and Invalid Arguments. **Quantified Statements:** Predicates and Quantified Statements, Statements with Multiple Quantifiers, Arguments with Quantified Statements.	12hrs	1	6	2

II	**Functions**: Functions Defined on General Sets, One-to-One and Onto, Inverse Functions, Composition of Functions, Cardinality with Applications to Computability. **Relations**:Relations on Sets, Reflexivity, Symmetry, and Transitivity, Equivalence Relations, Partial Order Relations **Recursion Relation**:Solving recurrence relations by iteration, Second order linear homogeneous recurrence relations with constant coefficients. general recursive definitions and structural induction.	12hrs	2		
III	**Counting Principles and probability:** Introduction, Possibility Trees and the Multiplication Rule, Counting Elements of Disjoint Sets: The Addition Rule, The Pigeonhole Principle, Counting Subsets of a Set: Combinations, r-Combinations with Repetition Allowed, Mathematical Induction, Strong Mathematical Induction and the Well-Ordering Principle for the Integers .Probability Axioms and Expected Value, Conditional Probability, Bayes' Formula, and Independent Events.	12hrs	2		
IV	**Graphs and Trees**: Definitions and Basic Properties, Trails, Paths, and Circuits, Matrix Representations of Graphs, Isomorphism of Graphs, Trees, Rooted Trees, Isomorphism of Graphs, Spanning trees and shortest paths.	12hrs	3		
V	**Elementary Number Theory and Methods of Proof**: Introduction to Direct Proofs, Rational Numbers, Divisibility, Division into Cases and the Quotient-Remainder Theorem, Floor and Ceiling, Indirect Argument: Contradiction and Contraposition, Two Classical Theorems, Applications in algorithms.	12hrs	4		

References;

1. Discrete Mathematics with Applications, Sussana S. Epp, 4th Edition, 2010
2. Discrete Mathematics, Schaum's Outlines Series, Seymour Lipschutz, Marc Lipson Tata MCGraw Hill 2007
3. Discrete Mathematics and its Applications , Kenneth H. Rosen , Tata MCGraw Hill
4. Discrete mathematical structures , B Kolman RC Busby, S Ross , PHI
5. Discrete structures , Liu , Tata MCGraw Hill

Course Description: BSc(Information Technology)	
Semester	**I**
Course Name	**Communication Skills**
Course Code	**UIT1CMT**
Eligibility for the Course	
Creit	**2**
Hours	**5Hrs per week**

Course Objectives	To understand the basics of Professional as well as Business Communication Skills.

Course Outcomes	**After completing the course, Student will be able to:**
	1) Elaborate examples on how to write business messages appropriately and propose the views in meetings and group discussions
	2) Develop Communication in different fields or departments.
	3) Design presentations and how to present one.

Module/ Unit	Course Description	Hrs.	CO No.	PSO No.	PO No.
I	**Understanding Business Communication:** Nature and Scope of Communication, process of communication, components **The Seven Cs of Effective** Communication: Completeness, Conciseness, Consideration, Concreteness, Clarity, Courtesy, Correctness **Non-verbal communication:** Personal appearance, facial expressions, movements, posture, gestures, eye contact, vocal communication techniques, voice, volume, pitch, rate of delivery, pronunciations, pause Communication barriers	12hrs	2	3	5
II	**Effective writing, reading skills, listening skills** **Writtng:** Notes making, Precis making **Reading skill:** mechanics of reading, guidelines to improving reading skills, types of readings, techniques of comprehension		1		

III	**Writing Business Messages and Documents:** Business writing, Business Correspondence, Instructions, Business Reports and Proposals, Career building, Job Application and Resume writing. **Group Communication:** Meetings and Conferences, Email correspondence Group Discussions and Team Presentations, Team Briefing, notices, agenda writing,	12hrs	1		
IV	**Understanding Specific Communication Needs:** Corporate Communication, Persuasive Strategies in Business Communication, Ethics in Business Communication, Business Communication Aids	12hrs	2		
v	**Presentation Skills:** Planning the presentations, executing the presentations, Impressing the audience by performing, Planning stage: Brainstorming, mind maps / concept maps, executing stage: chunking theory, creating outlines, Use of templates. Adding graphics to your presentation: Visual communication, Impress stage: use of font, colour, layout, Importance of practice and performance.	12hrs	3		

References:

1. Business Communication, Edited by Meenakshi Raman and Prakash Singh, Second Edition, Oxford University Press,
2. Professional Communication, ArunaKoneru, Tata McGraw Hill
3. Business Communication, Dr.Rishipal and Dr.JyotiSheoran, SPD
4. Strategies for improving your business communication, Prof. M. S. Rao, Shroff publishers and distributors
5. Graphics for Learning: Proven Guidelines for Planning, Designing, and Evaluating Visuals in Training Materials, Ruth C. Clark, Chopeta Lyons, Pfeiffer

Course Description: BSc(Information Technology)	
Semester	**I**
Course Name	**Environmental Studies**
Course Code	**USC1EVS**
Eligibility for the Course	
Credit	**2**
Hours	**30Hrs.**

Course Objectives	To develop environmental consciousness among the students

Course Outcomes	**After completing the course, Student will be able to:**
	1) Describe Environment
	2) Predict the consequences of human activities on the web of life
	3) Extend the values and responsibilities in solving current environmental problems and avoid future destruction

Module/ Unit	**Course Description**	Hrs.	CO No.	PSO No.	PO No.
1	**Introduction to Environmental Studies**	15hrs			
1.1	Definition, scope and importance of environmental studies, Need for Public awareness				
1.2	Ecosystem 1.2.1. Concept of ecosystem 1.2.2. Types of ecosystems, structure, characteristics and functions: Forest, Grassland, Desert, Aquatic ecosystem				
1.3	Biodiversity and its conservation 1.3.1. Introduction – Definition: genetic, species and ecosystem diversity. 1.3.2. Value of biodiversity: consumptive use, productive use, social, ethical, aesthetic and option values 1.3.3. Threats to biodiversity: habitat				

	loss, poaching of wildlife, man-wildlife conflicts. 1.3.4. Conservation of biodiversity: In-situ and Ex-situ conservation of biodiversity.				
2	**Environmental Pollution and Control**	15hrs			
2.1	Environmental pollution 2.1.1 Definition 2.1.2 Cause, effects and control measures of – a. Air pollution b. Water pollution c. Soil pollution d. Marine pollution e. Noise pollution f. Thermal pollution g. Nuclear hazards 2.1.3 Role of an individual in prevention of pollution.				
2.2	Environment Protection Act.: Brief introduction				

Course Description: B.Sc.(Information Technology)	
Semester	**I**
Course Name	**Introduction to C++ Programming Practical**
Course Code	**UIT1CPP**
Eligibility for the Course	
Credit	**2**
Hours	**3Hrs per week**

Course Objectives	The objective of this course is to introduce the concept of the basic programming language with C++.

Course Outcomes	**After completing the course, Student will be able to:**
	1) Develop Simple C++ Programs.
	2) Construct C++ programs using conditional statements and loops.
	3) Make use of functions in C++ programs.
	4) Build C++ Programs using Arrays.

Module/ Unit	**Course Description**	Hrs	CO No.	PSO No.	PO No.
1	**Basic Programs of C++** a. Write a program to display the message HELLO WORLD. b. Write a program to declare some variables of type int, float and double. Assign some values to these variables & display these values. c. Write a program to find the addition, subtraction, multiplication and division of two numbers**.**	3hrs	1	2	6

2	**Programs on variables** a. Write a program to swap to number using the third variable. b. Write a program to swap two numbers without using a third variable. c. Write a program to find the area of rectangle, square and circle. d. Write a program to find the volume of a cube, sphere, and cylinder.	3hrs	1		
3	**Conditional statements and loops(basic)** a. Write a program to enter a number from the user and display the month name. b. If number>13 then display invalid input using switch case. c. Write a program to check whether the number is even or odd. d. Write a program to check whether the number is positive, negative or zero. **e.** Write a program to find the smallest of three numbers.	3hrs	2		
4	**Conditional statements and loops(advanced)** a. Write a program to find the sum of squares of digits of a number. b. Write a program to reverse the digits of an integer. c. Write a program to find the sum of numbers from 1 to 100. **d.** Write a program to print the Fibonacci series.	3hrs	2		
5	**Programs on patterns** Programs on different patterns.	3hrs	1		
6	**Functions:** a. Programs on Functions. **b.** Write a program to demonstrate example of Inline Function	3hrs	3		

7	**Recursive functions** a. Write a program to find the factorial of a number using a recursive function. b. Write a program to find the sum of natural numbers using a recursive function.	3hrs	3		
8	**Arrays** a. Write a program to find the largest value that is stored in the array. b. Write a program to compute the sum of all elements stored in an array. c. Write a program to arrange the 'n' numbers stored in the array in ascending and descending order. **d.** Write a C++ program to rearrange a given sorted array of positive integers.	3hrs	4		
9	**String handling** a. String operations for string length , string concatenation b. String operations for string reverse, string comparison, **c.** Console formatting functions.	3hrs	4		
10	**Programs on Structures and Unions** a. Programs on structures. **b.** Programs on unions	3hrs	4		

References:

1. "Let us C++" , Y.P.Kanetkar, Seventh edition, BPB publication
2. "Problem Solving with C++" , Walter Savitch, Sixth Edition, Pearson Education.

Course Description: BSc(Information Technology)	
Semester	**I**
Course Name	**Digital Electronics Practical**
Course Code	**UIT1DEP**
Eligibility for the Course	
Credit	**2**
Hours	**3Hrs per week**

Course Objectives	The objective of this course is to acquire the basic knowledge of digital logic levels and the application of knowledge to understand digital electronics circuits. To prepare the learners to perform the analysis and design of various digital electronic circuits.

Course Outcomes	**After completing the course, Student will be able to:**
	1) Classify logic gates and their ICs and universal gates.
	2) Simplify the given Boolean expressions using a minimum number of logic gates and ICs.
	3) Build combinational circuits and code converters.
	4) Design Encoder, Decoder, Multiplexer and Demultiplexer

Module/ Unit	**Course Description**	Hrs	CO No.	PSO No.	PO No.
1	**1. Study of Logic gates and their ICs and universal gates:** a. Study of AND, OR, NOT, XOR, XNOR, NAND and NOR gates b. IC 7400, 7402, 7404, 7408, 7432, 7486, 74266 c. Implement AND, OR, NOT, XOR, XNOR using NAND gates. d.Implement AND, OR, NOT, XOR, XNOR using NOR gates.	3hrs	1	1	7

2	**Implement the given Boolean expressions using minimum number of gates.** a. Verifying De Morgan's laws. b. Implement other given expressions using minimum number of gates. c. Implement other given expressions using minimum number of ICs	3hrs	2		
3	**Implement combinational circuits.** a. Design and implement combinational circuit based on the problem given and minimizing using K-maps.	3hrs	3		
4	**Implement code converters.** a. Design and implement Binary – to – Gray code converter. b. Design and implement Gray – to – Binary code converter.Design and implement Binary – to – BCD code converter d. Design and implement Binary – to – XS-3 code converter	3hrs	3		
5	**Implement Adder and Subtractor Arithmetic circuits.** a. Design and implement Half adder and Full adder. b. Design and implement BCD adder. c. Design and implement XS – 3 adder. d. Design and implement binary subtractor. e. Design and implement BCD subtractor. f. Design and implement XS – 3 subtractor.	3hrs	3		

6	**Implement Arithmetic circuits.** a. Design and implement a 2-bit by 2-bit multiplier. b. Design and implement a 2-bit comparator.	3hrs	3		
7	**Implement Encode and Decoder and Multiplexer and Demultiplexers.** a. Design and implement 8:3 encoder. b. Design and implement 3:8 decoder. c. Design and implement 4:1 multiplexer. Study of IC 74153, 74157 d. Design and implement 1:4 demultiplexer. Study of IC 74139 e. Implement the given expression using IC 74151 8:1 multiplexer. f. Implement the given expression using IC 74138 3:8 decoder.	3hrs	4		
8	**Study of flip-flops and counters.** a. Study of IC 7473. b. Study of IC 7474. c. Study of IC 7476. d. Conversion of Flip-flops. e. Design of 3-bit synchronous counter using 7473 and required gates. f. Design of 3-bit ripple counter using IC 7473.	3hrs			

9	**Study of counter ICs and designing Mod-N counters.** a. Study of IC 7490, 7492, 7493 and designing mod-n counters using these. b. Designing mod-n counters using IC 7473 and 7400 (NAND gates)	3hrs			
10	**Design of shift registers and shift register counters.** a. Design serial – in serial – out, serial – in parallel – out, parallel – in serial – out, parallel– in parallel – out and bidirectional shift registers using IC 7474. b. Study of ID 7495. c Implementation of digits using seven segment displays.	3hrs			
11	**Study of Diode ROM Array.**	3hrs			
12	**Study of 555 Timer as an Astable multivibrator.**	3hrs			

References:

1. Digital Electronics and Logic Design, N. G. Palan, Technova
2. Modern Digital Electronics, R. P. Jain, Tata McGraw Hill, 3rd
3. Digital Principles and Applications, Malvino and Leach, Tata McGraw Hill

Course Description: B.Sc.(Information Technology)	
Semester	**I**
Course Name	**Operating Systems Practical**
Course Code	**UIT1OSP**
Eligibility for the Course	
Credit	**2**
Hours	**3Hrs per week**

Course Objectives	To make the learners familiar with the basics of different operating systems.

Course Outcomes	**After completing the course, Student will be able to:**
	1) Build virtual operating system.
	2) Demonstrate linux commands.
	3) Make use of utilities of windows, linux and cloud.
	4) Choose windows commands for file, folder creation.

Module/Unit	Course Description	Hrs	CO No.	PSO No.	PO No.
1.	**Installation of virtual machine software.**	3hrs	1	4	2
2.	**Installation of Linux operating system (RedHat / Ubuntu) on virtual machine.**	3hrs	1		
3.	**Installation of Windows operating system on virtual machine.**	3hrs	1		
4.	**Linux commands: Working with Directories:** a. pwd, cd, absolute and relative paths, ls, mkdir, rmdir, b. file, touch, rm, cp. mv, rename, head, tail, cat, tac, more, less, strings, chmod	3hrs	2		
5.	**Linux commands: Working with files** a. ps, top, kill, pkill, bg, fg, b. grep, locate, find, locate. c. date, cal, uptime, w, whoami, finger, uname, man, df, du, free, whereis, which. d. Compression: tar, gzip.	3hrs	2		

6.	**6. Windows (DOS) Commands – 1** a. Date, time, prompt, md, cd, rd, path. b. Chkdsk, copy, xcopy, format, fidsk, cls, defrag, del, move.	3hrs	3		
7.	**Windows (DOS) Commands – 2** a. Diskcomp, diskcopy, diskpart, doskey, echo b. Edit, fc, find, rename, set, type, ver **Working with different Cloud Platform** a. Google Drive b. Amazon web service	3hrs	3		
8.	**8. Working with Windows Desktop and utilities** a. Notepad b. Wordpad c. Paint d. Taskbar e. Adjusting display resolution f. Using the browsers g. Configuring simple networking h. Creating users and shares	3hrs	4		
9.	**Working with Linux Desktop and utilities** a. The vi editor. b. Graphics c. Terminal d. Adjusting display resolution e. Using the browsers f. Configuring simple networking g. Creating users and shares	3hrs	4		
10.	**Installing utility software on Linux and Windows.** **a) Protection in File System.**	3hrs	4		

References

1. UNIX Concepts and Applications- Sumitabha Das, 4th Edition , Tata McGraw Hill Publishers

Course Description: B.Sc.(Information Technology)	
Semester	**I**
Course Name	**Discrete Mathematics Practical**
Course Code	**UIT1DMP**
Eligibility for the Course	
Credit	**2**
Hours	**3Hrs per week**

Course Objectives	To familiarize the students with the fundamental concepts of scilab and develop programming skill to effectively implement for problems.

Course Outcomes	**After completing the course, Student will be able to:**
	1) Make use of the basic commands of scilab.
	2) Construct a formula for recurrence relation, counting and probability using scilab.
	3) Analyse the concept of properties of integers and operations using scilab.
	4) Examine the properties of graphs, applications of graphs and trees.

Module/ Unit	Course Description	Hrs.	CO No.	PSO No.	PO No.
1	**Set Theory** a. Inclusion Exclusion principle. b. Power sets c. Mathematical Induction	3hrs	1	6	2
2	**Functions and Algorithms** a. Recursively defined functions b. Cardinality c. Polynomial evaluations d. Greatest Common Divisor	3hrs	1		
3	**Recurrence Relation** a. Linear homogeneous recurrence relations with constant coefficients b. Solving linear homogeneous recurrence relations with constant coefficients c. Solving general homogeneous linear recurrence relations.	3hrs	2		

4	**Counting :** a. Sum rule principle b. Product rule principle c. Factorial d. Binomial coefficients e. Permutations f. Permutations with repetitions g. Combinations h. Combinations with repetitions i. Ordered partitions j. Unordered partitions	3hrs	2		
5	**Probability Theory:** a. Sample space and events b. Finite probability spaces c. Equiprobable spaces d. Addition Principle e. Conditional Probability f. Multiplication theorem for conditional probability. g. Independent events h. Repeated trials with two outcomes.	3hrs	2		
6	**Graph Theory** a. Paths and connectivity b. Minimum spanning tree c. Isomorphism	3hrs	4		
7	**Direct Graphs** a. Adjacency matrix b. Path matrix	3hrs	4		
8	**Properties of integers** a. Division algorithm b. Primes c. Euclidean algorithm d. Fundamental theorem of arithmetic e. Congruence relation f. Linear congruence equation	3hrs	3		
9	**Algebraic Systems** a. Properties of operations	3hrs	3		
10	**Boolean Algebra** a. Basic definitions in Boolean Algebra b. Boolean algebra as lattices	3hrs	3		

References

1. Scilab textbook companion for discrete mathematics by S. Lipschutz, M. Lipson and V. H. Patil

Course Description B.Sc. (Information Technology)	
Semester	**I**
Course Name	**Communication Skills Practical**
Course Code	**UIT1CSP**
Eligibility for the Course	
Credit	**2**
Hours	**3Hrs per week**

Course Objectives	To understand the basics of Professional as well as Business Communication Skills.

Course Outcomes	**After completing the course, Student will be able to:**
	1) Develop pronunciation skills, listening skills, writing skills
	2) Construct storytelling, advertising, role plays and situational conversations
	3) Take part in interviews and group discussions
	4) Build presentations

Module/ Unit	Course Description	Hrs.	CO No.	PSO No.	PO No.
I	**1. Mock Interviews** a. You are in the HR department of an organization and you are supposed to hire a Candidate. Write a telephonic interview between you and the client. b. Write an interview (a face-to-face interview) between Mr Smith and Mr John regarding a Job position in a company. Mr John is the manager whereas Mr Smith is the candidate. c. You are asked to conduct a video interview for hiring a candidate in your company. Write the questions than you can ask and possible answers that can be given by the Candidate.	3hrs	1	3	4
II	**Presentations** a. 3D Glasses b. Apple Talk c. Mobile IP	3hrs	1		

	d. Big Data				
III	**Group Discussions** a. Impact of covid 19 on education b. Gender Equality c. India: The super Powering country d. Social Media: Impact on human behaviour and society	3hrs	3		
IV	**Role Play** a. Assume that you are a Sales person. Write a conversation between the sales person and the customer for selling a laptop. b. Introduce yourself as an electronic gadget c. Assume that you are a king of a kingdom. Write a conversation between you and your ministers regarding the development of your kingdom.	3hrs	2		
v	**Situational Conversion** a. Tell me about a time you proved you're the perfect person for this job. b. What would you do if you made a mistake that no one else noticed? Would you address the error and risk slowing things down or ignore it to keep the project or task moving forward? c. What would you do if you were asked to perform a task you've never done before? d. What would you do if an angry and 12hrs dissatisfied customer confronted you? How would you resolve their concern?	3hrs	2		
VI	**Advertising** a. There is a campaign in you college regarding women's safety in college. Write an advertisement for the same. b. A new product named 'Techno' is introduced by an IT company which helps you locate your personal things like mobile phone, wallet, keys, etc. Write an advertisement to sell this product. Also state its features. c. There is a new TV reality show and you are asked to promote it. How will you write an advertisement for the same?	3hrs	2		

VII	**Story-Telling** a. A middle-aged woman discovers a ghost. b. A group of children discover a dead body c. A long journey is interrupted by a disaster.	3hrs	2		
VIII	**Pronunciation Skills**	3hrs	1		
IX	**Listening Skills**	3hrs	1		
X	**Writing Skills**	3hrs	1		

References:

1. Business Communication, Edited by Meenakshi Raman and Prakash Singh, Second Edition, Oxford University Press,
2. Professional Communication, ArunaKoneru, Tata McGraw Hill
3. Business Communication, Dr.Rishipal and Dr.JyotiSheoran, SPD
4. Strategies for improving your business communication, Prof. M. S. Rao, Shroff publishers and distributors
5. Graphics for Learning: Proven Guidelines for Planning, Designing, and Evaluating Visuals in Training Materials, Ruth C. Clark, Chopeta Lyons, Pfeiffer

Semester II

Course Description: BSc(Information Technology)	
Semester	**II**
Course Name	**Object Oriented Programming**
Course Code	**UIT2OPT**
Eligibility for the Course	
Credit	**2**
Hours	**5Hrs per week**

Course Objectives	To learn advanced features of the C++ programming language as a continuation of the previous course, to learn the characteristics of an object-oriented programming language: data abstraction and information hiding, inheritance, and dynamic binding of the messages to the methods.

Course Outcomes	**After completing the course, Student will be able to:**
	1) Explain characteristics of object oriented programming approach with C++.
	2) Make use of operators in C++.
	3) Evaluate the concept of Template, Strings, Streams
	4) Utilize different file handling features

Module/ Unit	Course Description	Hrs.	CO No.	PSO No.	PO No.
I	**Introduction to OOPs:** Need object oriented programming, comparison of procedural and object oriented approach, characteristics of OOPs – object, classes, polymorphism, inheritance, reusability, data hiding and abstraction, applications of OOPs. **Classes and Objects:** Class declaration, constructors, constructor initialization lists, access functions, private member functions, the copy constructor, the class destructor ,constant objects, structures, pointers to objects, static data members,	12hrs	1	7	2

	static function members				
II	**Operator Overloading:**, overloading the assignment operator, the this pointer, overloading arithmetic operators, overloading the arithmetic assignment, operators, overloading the relational operators, overloading the stream operators, conversion operators ,overloading the increment and decrement operators, overloading the subscript operator	12hrs	2		
III	**Composition and Inheritance:** Type of inheritance, protected class members, overriding and dominating inherited members, Function Overloading, private access verses protected access, virtual functions and polymorphism, pure virtual function, virtual destructors, abstract base classes **File Handling:** Classes for file stream operations, opening and closing a file, detecting end of file, file modes, file pointers and their manipulations, sequential input and output operations, random access, file operations error handling, command line argument	12hrs	4		
III	**Strings and Streams:** the string class interface, the constructors and destructor , the copy constructor, the assignment operator, the addition operator , an append operator, access functions , the comparison operators, stream operators, Stream classes, the ios class, ios format flags, ios state, variables, the istream and ostream classes, unformatted input functions, unformatted output functions, stream manipulators. **Exception Handling:** Introduction, Basics of Exception Handling, Exception Handling mechanism, Throwing Mechanism, Catching Mechanism, Specifying Exception	12hrs	3		
V	**Templates and Iterators:** function templates, class templates, container classes, subclass templates, passing	12hrs	3		

	template classes to template parameters, iterator classes **Libraries:** the standard C++ library, proprietary libraries, contents of the standard c headers, string streams, file processing, the standard template library				

References

1) Object Oriented Design by Rumbaugh (Pearson publication)
2) Object-oriented programming in Turbo C++ By Robert Lafore, Galgotia Publication.
3) Object-oriented programming with C++ by E.Balagurusamy, 2nd Edition, TMH.

Course Description :BSc(Information Technology)	
Semester	**II**
Course Name	**Microprocessor Architecture**
Course Code	**UIT2MAT**
Eligibility for the Course	
Credit	**2**
Hours	**5Hrs per week**

Course Objectives	The objective of this course is to introduce the basic structure of 8085 Microprocessor, Assembly Language Programming techniques and its instruction set.

Course Outcomes	**After completing the course, Student will be able to:**
	1) Explain Microprocessor, Microcomputer and Assembly Language.
	2) Elaborate the concepts of Microprocessor Architecture, Interface Devices and Assembly Language.
	3) Make use of counter and time delay, Stack and Subroutines.
	4) Explain 8086 microprocessor, Software Development Systems and Interrupt.

Module/ Unit	Course Description	Hrs.	CO No.	PSO No.	PO No.
I	**Microprocessor, microcomputers, and Assembly Language:** Microprocessor, Microprocessor Instruction Set and Computer Languages, From Large Computers to Single-Chip Microcontrollers, Applications. **Microprocessor Architecture and Microcomputer System:** Microprocessor Architecture and its operations, Memory, I/O Devices, Microcomputer System, Logic Devices and Interfacing, Microprocessor-Based System Application. **Introduction to 8085 Assembly Language Programming:** The 8085 Programming Model, Instruction Classification, Instruction, Data and Storage, Writing assembling and Execution of a simple program, Writing and Assembling Program.	12hrs	1	2	3

II	**Introduction to 8085 Instructions:** Overview of 8085 Instruction Set, Data Transfer Operations, Arithmetic Operations, Logic Operation, Branch Operation, Writing Assembly Languages Programs, Debugging a Program **8085 Microprocessor Architecture and Memory Interface:** Introduction, 8085 Microprocessor unit, 8085-Based Microcomputer, Memory Interfacing, Interfacing the 8155 Memory Segment, Illustrative Example: Designing Memory for the MCTS Project, Testing and Troubleshooting Memory Interfacing Circuit, 8085-Based Single-Board microcomputer. **Interfacing of I/O Devices:** Basic Interfacing concepts, Interfacing Output Displays, Interfacing Input Devices, Memory Mapped I/O, Testing and Troubleshooting I/O Interfacing Circuits.	12hrs	2		
III	**Programming Techniques With Additional Instructions:** Programming Techniques: Looping, Counting and Indexing, Additional Data Transfer and 16-Bit Arithmetic Instructions, Arithmetic Instruction Related to Memory, Logic Operations: Rotate, Logics Operations: Compare, Dynamic Debugging. **Counters and Time Delays:** Counters and Time Delays, Illustrative Program: Hexadecimal Counter, Illustrative Program: zero-to-nine (Modulo Ten) Counter, Generating Pulse Waveforms, Debugging Counter and Time-Delay Programs.	12hrs	3		

IV	**Stacks and Sub-Routines:** Stack Subroutine, Restart, Conditional Call, Return Instructions, Advanced Subroutine concepts. **8086 Microprocessor:** Introduction to 8086 Microprocessor, Intel Microprocessor families, 8086 Microprocessor Architecture, Register Organization, Pin Description, Modes of Operation, Difference between 8085 instructions and 8086 instructions.	12hrs	3 4		
V	**Software Development System and Assemblers:** Microprocessors-Based Software Development system, Operating System and Programming Tools, Assemblers and Cross-Assemblers, Writing Program Using Cross Assemblers. **Interrupts:** The 8085 Interrupt, 8085 Vectored Interrupts, Restart as S/W Instructions, Additional I/O Concepts and processes.	12hrs	4		

References

1) Microprocessors Architecture, Programming and Applications with the 8085, Ramesh Gaonkar, Fifth Edition, PENRAM
2) Computer System Architecture, M. Morris Mano, PHI
3) Structured Computer Organization, Andrew C. Tanenbaum, PHI

Course Description: BSc(Information Technology)	
Semester	**II**
Course Name	**Database Management System**
Course Code	**UIT2DMT**
Eligibility for the Course	
Credit	**2**
Hours	**5Hrs per week**

Course Objectives	The objective of this course is to introduce the concept of the DBMS with respect to the relational model, to specify the functional and data requirements for a typical database application and to understand creation, manipulation and querying of data in databases and the basic of PL/SQL

Course Outcomes	**After completing the course, Student will be able to:**
	1) Design E-R model to represent database
	2) Design the database with normalization
	3) Explain the fundamental of RDBMS
	4) Explain the transactions of database and basic of PL/SQL

Module/ Unit	Course Description	Hrs.	CO No.	PSO No.	PO No.
I	**Introduction**: What is database system, purpose of database system, view of data, relational databases, database architecture, transaction management, **Data Models** :The importance of data models, Basic building blocks ,Business rules, The evolution of data models, Degrees of data abstraction	12hrs	1	2	3
II	**Database Design, ER-Diagram** Database design and ER Model: overview, ER-Model, Constraints, ER Diagrams,ERD Issues, weak entity sets, Codd's rules, Relational Schemas, Logical view of data, keys, integrity rules. **Relational Algebra and Calculus**: Relational algebra: introduction, Selection and projection, set operations, renaming, Joins, Division, syntax, semantics. Operators, grouping and ungrouping, relational comparison.	12hrs	3		
III	**Constraints, Views and SQL** :What is constraints, types of constrains,Integrity constraints, Views: Introduction to views, data independence,security, updates on views, comparison between tables and views **SQL**:data definition, aggregate function,single row function Null Values, nested sub queries,Joined relations,.	12hrs	4		
IV	**Transaction management and Concurrency control**: Transaction management: ACID properties, serializability and concurrency control,Lock based concurrency control (2PL, Deadlocks),Time stamping methods, optimistic methods, database recovery management.	12hrs	4		

V	**PL-SQL:** Beginning with PL / SQL, Identifiers and Keywords, Operators, Expressions, Sequences, Overview and benefits of PL/SQL, Subprograms, types of PL/SQL blocks, Simple Anonymous Block, Identifiers, types of Identifiers, Declarative Section, variables, Scalar Data Types, The %TYPE Attribute, Executable Statements, PL/SQL Block Syntax, Comment the Code, Convert Data Types, Nested Blocks, Operators. Invoke SELECT Statements in PL/SQL, Save and Discard Transactions. .	12hrs	4		

References

1) "Database System and Concepts",A Silberschatz, H Korth, S Sudarshan, , fifth Edition McGraw-Hill .
2) "Database Systems", Rob, Coronel, Seventh Edition, Cengage Learning

Course Description : B.Sc. (Information Technology)	
Semester	**II**
Course Name	**Numerical Methods**
Course Code	**UIT2NMS**
Eligibility for the Course	
Credit	**2**
Hours	**5Hrs per week**

Course Objectives	The course is designed to have a grasp of important concepts of Numerical Methods in a scientific way. The learner is expected to solve as many examples as possible to get complete clarity and understanding of the topics covered.

Course Outcomes	**After completing the course, Student will be able to:**
	1) Solve algebraic, transcendental and simultaneous systems of equations using numerical methods.
	2) Evaluate the functions and their derivatives using interpolation.
	3) Solve differential equations and integration by using numerical methods.
	4) Apply triangularization method, LU decomposition, cholesky method, power and inverse power method.

Module/ Unit	Course Description	Hrs	CO No.	PSO No.	PO No.
I	**Solutions of Algebraic and Transcendental Equations**: The Bisection Method, The Newton-Raphson Method, The Regula-falsi method, The Secant Method**.** **Solution of simultaneous algebraic equation (linear) using iterative methods:** Gauss Elimination Method, Gauss Jordan Method, Gauss Jacobi Method, Gauss Seidel Method	12hrs	1		
II	**Interpolation:** Forward Difference, Backward Difference, Central Differences, Different Types of Operators, Relation between Operators, Newton's Forward Difference Interpolation, Newton's Backward Difference Interpolation, Divided Differences, Newton's Divided Difference Interpolation, Lagrange's	12hrs	2		

	Interpolation, Spline Interpolation.				
III	**Numerical differentiation:** Numerical differentiation, Methods based on finite Differences: Derivatives using Newton's Forward Difference Interpolation, Newton's Backward Difference Interpolation, Newton's Divided Difference Interpolation & Lagrange's Interpolation. **Numerical solution of 1st and 2nd order differential equations:** Taylor series, Picard's Method, Euler's Method, Modified Euler's Method, Runge-Kutta Method for 1st and 2nd Order Differential Equations.	12hrs	2,3		
IV	**Numerical integration:** Trapezoidal Rule, Simpson's 1/3rd and 3/8th rules, Romberg Method, Gauss Legendre Integration Method, Gauss Chebyshev Integration Method. **Double Integration:** Trapezoidal Method, Simpson's Method	12hrs	3		
V	**Linear System of equations Direct Method:** Triangularization Method, LU Decomposition, Cholesky Method, Partition Method. **Eigen value Problem** : Power Method	12hrs	4		

References:

1) Numerical Methods for Scientific and Engineering Computation, M. K. Jain, S. R. K. Iyengar and R. K. Jain, New age International Publishers, Fourth Edition, 2003
2) Introductory Methods of Numerical Methods , S. S. Shastri , PHI , Vol – 2
3) Numerical Methods for Engineers , Steven C. Chapra, Raymond P. Canale , Tata Mc Graw Hill , 6th Edition , 2010
4) Numerical Analysis , Richard L. Burden, J. Douglas Faires , Cengage Learning , 9th Edition, 2011
5) Numerical and Statistical Technique, QaziShoeb Ahmad, Zubair Khan, Shadad Ahmad Khan, Ane's Student Edition

Course Description:BSc(Information Technology)	
Semester	**II**
Course Name	**Web Programming**
Course Code	**UIT2WPT**
Eligibility for the Course	
Credit	**2**
Hours	**5Hrs per week**

Course Objectives	On completion of this course, a learner will be able to develop a web application using web technologies. Learners will gain the skills and project-based experience needed for entry into web application and development careers. Learners will be able to develop a dynamic webpage by the use of java script , jquery, xml, basic php along with interaction with mysql database.

Course Outcomes	**After completing the course, Student will be able to:**
	1) Illustrate the HTML5 tags used to develop static web pages.
	2) Make use of CSS to improve the look and feel of web pages.
	3) Elaborate the creation of dynamic web pages using server side PHP programming and Database connectivity.
	4) Explain javascript event handling and functions.

Module/ Unit	Course Description	Hrs.	CO No.	PSO No.	PO No.
I	**Internet and the World Wide Web:** What is Internet? Introduction to internet and its applications, E-mail, telnet, FTP, e-commerce, video conferencing, e-business. Internet service providers, domain name server, internet address, World Wide Web (WWW): World Wide Web and its evolution, uniform resource locator (URL) **HTML5:** Introduction, Why HTML5? Formatting text by using tags, using lists and backgrounds, Creating hyperlinks and anchors. Style sheets, CSS formatting text using style sheets, formatting paragraphs using style sheets, Creating image map, redirecting to another URL	12hrs	2	2	3
II	**HTML5 Tables, Forms and Media:** Creating tables: creating simple table, specifying the size of the table, specifying the width of the column, merging table cells, Creating user forms: creating basic form, using check boxes and option buttons, creating lists, additional input types in HTML5 **Bootstrap:** Introduction, Why Use Bootstrap?, Create First Web Page With Bootstrap, Bootstrap Grids, Bootstrap Typography Classes, Bootstrap Tables, Bootstrap Images, Bootstrap Alerts, Bootstrap Buttons, Bootstrap Progress Bars, Bootstrap basic Forms	12hrs	1		

III	**Java Script:** Introduction, Client-Side JavaScript, Server-Side JavaScript, JavaScript Objects, JavaScript Security. **Operators:** Assignment Operators, Comparison Operators, Arithmetic Operators, % (Modulus), ++(Increment), --(Decrement), -(Unary Negation), Logical Operators, Short-Circuit Evaluation, String Operators, Special Operators, ?: (Conditional operator), , (Comma operator), delete, new, this, void **Statements:** Break, comment, continue, delete, do...while, export, for, for...in, function, if...else, import, labelled, return, switch, var, while, with. **Core JavaScript (Properties and Methods of Each):** Array, Boolean, Date, Function, Math, Number, Object, String, regExp **Document and its associated objects:** Document, Link, Area, Anchor, Image, Applet, Layer **Events and Event Handlers:** General Information about Events, Defining Event Handlers, event, onAbort, onBlur, onChange, onClick, onDblClick, onDragDrop, onError, onFocus, onKeyDown, onKeyPress, onKeyUp, onLoad, onMouseDown, onMouseMove, onMouseOut, onMouseOver, onMouseUp, onMove, onReset, onResize, onSelect, onSubmit, onUnload	12hrs	4		

IV	**PHP:** Why PHP and MySQL? Server-side scripting, PHP syntax and variables, comments, types, control structures, branching, looping, termination, functions, passing information with PHP, GET, POST, formatting form variables, superglobal arrays, strings and string functions, regular expressions, arrays, number handling, basic PHP errors/problems. **XML:** Comparing XML with HTML,Advantages and Disadvantages of XML,Structure of an XML Document,XML entity references,DTD,XSLT.	12hrs	3		
V	**Advanced PHP and MySQL:** PHP/MySQL Functions, Integrating web forms and databases, Displaying queries in tables, Building Forms from queries, String and Regular Expressions, Sessions, Cookies and HTTP, E-Mail **Introduction to JQuery:** Fundamentals, Selectors, Methods to access HTML attributes.	12hrs	3		

References

1) "HTML5 Step by Step", FaitheWempen , Microsoft Press
2) "JavaScript 2.0: The Complete Reference ",Thomas Powell and Fritz Schneider , Second Edition, Tata McGraw Hill Publication
3) "PHP 5.1 for Beginners" , Ivan Bayross , Sharanam Shah, SPD Publication
4) "PHP 6 and MySQL Bible ",SteveSuehring, Tim Converse, Joyce Park, Wiley Publication
5) "PHP Project for Beginners ",SharanamShah, Vaishali Shah , SPD Publication
6) "Web Design The Complete Reference " , Thomas Powell , Tata McGraw Hill Publication
7) "Head First HTML 5 programming ",Eric Freeman , O'Reilly Publication

Course Description: BSc(Information Technology)	
Semester	**II**
Course Name	**Spoken English**
Course Code	**USC2CSK**
Eligibility for the Course	
Credit	**2**
Hours	**30Hrs.**

Course Objectives	To enhance communication skills of the students

Course Outcomes	**After completing the course, Student will be able to:**
	1) Develop an understanding of communication skills to face challenges of real and corporate life
	2) Show enhancement in the communication skill
	3) Demonstrate Leadership qualities, team-work, decision making

Module/ Unit	Course Description	Hrs.	CO No.	PSO No.	PO No.
I	**Academic Skills :** **1.1 Essentials of Grammar**: Parts of speech, Articles, Modals, Sentences and their types., Punctuation marks **1.2 Employment Communication**: Introduction, Resume, Curriculum Vitae, Scannable Resume, Developing an Impressive Resume, Formats of Resume, Job Application or Cover Letter. Email Writing **1.3 Professional Presentation**: Nature of Oral Presentation, planning a Presentation, Preparing the Presentation, Delivering the Presentation **1.4 Job Interviews**: Introduction, Importance of Resume, Definition of Interview, Background Information, Types of Interviews,	15hrs			

	Preparatory Steps for Job Interviews, Interview Skill Tips, Changes in the Interview Process, FAQ During Interviews **1.5** **Group Discussion**: Introduction, Ambience/Seating Arrangement for Group Discussion, Importance of Group Discussions, Difference between Group Discussion, Panel Discussion and Debate, Traits, Types of Group Discussions, topic based and Case based Group Discussion, Individual Traits				
II	**Soft and Professional Skills:** **2.1** Introduction to Soft Skills and Hard Skills **2.2** **Personality Development**: Knowing Yourself, Positive Thinking, Johari's Window, Communication Skills, Non-verbal Communication, Physical Fitness Definition **2.3** **Etiquette and Mannerism**: Introduction, Professional Etiquette, Technology Etiquette **2.4** **Communication Techniques:** **2.5** **Ethical Values**: Ethics and Society, Theories of Ethics, Correlation, between Values and behaviour, Nurturing Ethics, Importance of Work Ethics, Problems in the Absence of Work Ethics **2.6** **Leadership and Team Building**: Leader and Leadership, Leadership Traits, Culture and Leadership, Leadership Styles and Trends, Team Building, Types of Teams	15hrs	1		

Course Description: BSc(Information Technology)	
Semester	**II**
Course Name	**Object Oriented Programming Practical**
Course Code	**UIT2OPP**
Eligibility for the Course	
Credit	**2**
Hours	**3Hrs per week**
Course Description: BSc(Information Technology)	

Course Objectives	Main objective to teach learners the basic concepts and technique which form the object oriented programming paradigm

Course Outcomes	**After completing the course, Student will be able to:**
	1) Construct program using classes, constructors, inheritance
	2) Design programs using virtual functions and abstract classes
	3) Build program using operator overloading, template, exceptional handling and string handling
	4) Utilize private function, friend function.

Module/ Unit	Course Description	Hrs.	CO No.	PSO No.	PO No.
1	**Classes and methods:** a. Design an employee class for reading and displaying the employee information,the getInfo() and displayInfo() methods will be used repectively. Where getInfo() will be private method b. Design the class student containing getData() and displayData() as two of its methods which will be used for reading and displaying the student information respectively. Where getData() will be private method. Design the class Demo which will contain the following methods: readNo(), factorial() for calculating the factorial of a number, reverseNo() will reverse the given number, isPalindrome() will check the given number is palindrome, isArmstrong() which will calculate the given number is armStrong or not.WhereareadNo() will be private method. d. Write a program to demonstrate function definition outside class and accessing class members in function definition.	3hrs	1		
2	**Using friend functions:** a. Write a friend function for adding the two complex numbers, using a single class. b. Write a friend function for adding the two different distances and display its sum, using two classes. c. Write a friend function for adding the two matrix from two different classes and display its sum.	3hrs	4		

3	**Constructors and method overloading:** a. Design a class Complex for adding the two complex numbers and also show the use of constructor. b. Design a class Geometry containing the methods area() and volume() and also overload the area() function . c. Design a class StaticDemo to show the implementation of static variable and static function.	3hrs	1		
4	**Operator Overloading:** a.Overload the operator unary(-) for demonstrating operator overloading. b. Overload the operator + for adding the timings of two clocks,And also pass objects as an argument. c.Overload the + for concatenating the two strings. For e.g “Py” + “thon” = Python.	3hrs	3		
5	**Inheritance:** a.Design a class for single level inheritance using public and private type derivation. b.Design a class for multiple inheritances. c.Implement the hierarchical inheritance d.Design a class for multilevel inheritances.	3hrs	1		

6	**Virtual functions and abstract classes** a.Implement the concept of method overriding. b.Show the use of virtual function c.Show the implementation of abstract class.	3hrs	2		
7	**String handling** a..String operations for string length , string concatenation b.String operations for string reverse, string comparison, c. Console formatting functions	3hrs	3		
8	**Exception handling** a.Show the implementation of exception handling b.Show the implementation for exception handling for strings c.Write a program to demonstrate how we can restrict a function to throw only certain types of exceptions and not all.	3hrs	3		
9	**File handling** a.Design a class FileDemo open a file in read mode and display the total number of words and lines in the file. b. Design a class to handle multiple files and file operations. c. Design a editor for appending and editing the file.	3hrs	3		

10	**Templates** Design the template class library for concatenating two strings b.Design the implementation of template class library for swap function. c.Design the template class library for sorting ascending to descending and vice-versa	3hrs	3		

References

1. Object-oriented programming in Turbo C++ By Robert Lafore, Galgotia Publication.
2. Object-oriented programming with C++ by E.Balagurusamy, 2nd Edition, TMH.

Course Description: BSc(Information Technology)	
Semester	**II**
Course Name	**Microprocessor Architecture Practical**
Course Code	**UIT2MAP**
Eligibility for the Course	
Credit	**2**
Hours	**3Hrs. per week**

Course Objectives	The objective of this course is to create simple assembly language programming and Operations on Memory Locations.

Course Outcomes	**After completing the course, Student will be able to:**
	1) Create simple Assembly Language Programs.
	2) Evaluate operations on memory locations.
	3) Develop packing and unpacking operations.
	4) Make use of register operations.

Module/Unit	**Course Description**	Hrs.	CO No.	PSO No.	PO No.
1	**Perform the following Operations related to memory locations.** a. Store the data byte 32H into memory location 4000H. b. Exchange the contents of memory locations 2000H and 4000H	3hrs	2	2	7
2	**Assembly language programs.** a. Subtract the contents of memory location 4001H from the memory location 2000H and place the result in memory location 4002H.	3hrs	1		

	b. Subtract two 8-bit numbers. c. Add the 16-bit number in memory locations 4000H and 4001H to the 16-bit number in memory locations 4002H and 4003H.The most significant eight bits of the two numbers to be added are in memory locations 4001H and 4003H.Store the result in memory locations 4004H and 4005H with the most significant byte in memory location 4005H. d. Add the contents of memory locations 40001H and 4001H and place the result in the memory locations 4002Hand 4003H. e.Write a program using 8085 Microprocessor for Decimal, Hexadecimal addition and subtraction of two Numbers. f. Find the l's complement of the number stored at memory location 4400H and store the complemented number at memory location 4300H. g. Write a program to convert given Hexadecimal number into its equivalent ASCII number and vice versa using 8085 instruction set.				

3	**Packing and unpacking operations.** a. Pack the two unpacked BCD numbers stored in memory locations 4200H and 4201H and store result in memory location 4300H.Assume the least significant digit is stored at 4200H. b. Two digit BCD number is stored in memory location 4200H. Unpack the BCD number and store the two digits in memory locations 4300H and 4301H such that memory location 4300H will have lower BCD digit.	3hrs	3		
4	**Register Operations.** a. Write a program to shift an eight bit data four bits right. Assume that data is in register C b. Program to shift a 16-bit data 1 bit left. Assume data is in the HL register pair c. Write a set of instructions to alter the contents of flag register in 8085. d. Write a program to count number of l's in the contents of D register and store the count in the B register.	3hrs	4		

5	**Multiple memory locations.** a. Calculate the sum of series of numbers. The length of the series is in memory location 4200H and the series begins from memory location 4201H. a. Consider the sum to be 8 bit number. So, ignore carries. Store the sum at memory location 4300H.Consider the sum to be 16 bit number. Store the sum at memory locations 4300H and 4301H b. Multiply two 8-bit numbers stored in memory locations 2200H and 2201H by repetitive addition and store the result in memory locations 2300H and 2301H. c. Find the largest number in a block of data. The length of the block is in memory location 2200H and the block itself starts from memory location 2201H. Store the maximum number in memory location 2300H. Assume that the numbers in the block are all 8 bit unsigned binary numbers.	3hrs	1		
6	**Calculations with respect to memory locations.** a. Write a program to sort given 10 numbers from memory location 2200H in the ascending order. b. Calculate the sum of series of even numbers from the list of numbers. The length of the list is in memory location 2200H and the series itself begins from memory location 2201H. Assume the sum to be 8 bit number so you can ignore carries and store the sum at	3hrs	1		

	memory location 2Sample problem: c. Calculate the sum of series of odd numbers from the list of numbers. The length of the list is in memory location 2200H and the series itself begins from memory location 2201H. Assume the sum to be 16-bit. Store the sum at memory locations 2300H and 2301H. d. Find the square of the given numbers from memory location 6100H and store the result from memory location 7000H e. Search the given byte in the list of 50 numbers stored in the consecutive memory locations and store the address of memory location in the memory locations 2200H and 2201H. Assume byte is in the C register and starting address of the list is 2000H. If byte is not found store 00 at 2200H and 2201H f. Add 2 arrays having ten 8-bit numbers each and generate a third array of result. It is necessary to add the first element of array 1 with the first element of array-2 and so on. The starting addresses of array l, array2 and array3 are 2200H, 2300H and 2400H, respectively				

7	**Assembly programs on memory locations.** a. Write an assembly language program to separate even numbers from the given list of 50 numbers and store them in the another list starting from 2300H. Assume starting address of 50 number list is 2200H b. Add even parity to a string of 7-bit ASCII characters. The length of the string is in memory location 2040H and the string itself begins in memory location 2041H. Place even parity in the most significant bit of each character. c .A list of 50 numbers is stored in memory, starting at 6000H. Find number of negative, zero and positive numbers from this list and store these results in memory locations 7000H, 7001H, and 7002H respectively d. Write an assembly language program to generate fibonacci number. e. Program to calculate the factorial of a number between 0 to 8.	3hrs	1		

8	**String operations in assembly programs.** a. Write an 8085 assembly language program to insert a string of four characters from the tenth location in the given array of 50 characters b. Write an 8085 assembly language program to delete a string of 4 characters from the tenth location in the given array of 50 characters. c. Multiply the 8-bit unsigned number in memory location 2200H by the 8-bit unsigned number in memory location 2201H. Store the 8 least significant bits of the result in memory location 2300H and the 8 most significant bits in memory location 2301H. d.Divide the 16-bit unsigned number in memory locations 2200H and 2201H (most significant bits in 2201H) by the B-bit unsigned number in memory location 2300H store the quotient in memory location 2400H and remainder in 2401H e. DAA instruction is not present. Write a subroutine which will perform the same task as DAA	3hrs	1	2	

9	**Calculations on memory locations.** a. To test RAM by writing '1' and reading it back and later writing '0' (zero) and reading it back. RAM addresses to be checked are 40FFH to 40FFH. In case of any error, it is indicated by writing 01H at port 10 b. Arrange an array of 8 bit unsigned no in descending order c. Transfer ten bytes of data from one memory to another memory block. Source memory block starts from memory location 2200H whereas destination memory block starts from memory location 2300H d. Write a program to find the Square Root of an 8 bit binary number. The binary number is stored in memory location 4200H and store the square root in 4201H. e. Write a simple program to Split a HEX data into two nibbles and store it in memory	3hrs	2	1	

10	**Operations on BCD numbers.** a. Add two 4 digit BCD numbers in HL and DE register pairs and store result in memory locations, 2300H and 2301H. Ignore carry after 16 bit. b. Subtract the BCD number stored in E register from the number stored in the D register c. Write an assembly language program to multiply 2 BCD numbers .	3hrs	4	3	

References

1) Microprocessors Architecture, Programming and Applications with the 8085, Ramesh Gaonkar, Fifth Edition, PENRAM
2) Computer System Architecture, M. Morris Mano, PHI
3) Structured Computer Organization, Andrew C. Tanenbaum, PHI

Course Description: BSc(Information Technology)	
Semester	**II**
Course Name	**Database Management System Practical**
Course Code	**UIT2DSP**
Eligibility for the Course	
Credit	**2**
Hours	**3Hrs per week**

Course Objectives	To give a good formal foundation on the relational model of data, give an introduction to systematic database design approaches

Course Outcomes	**After completing the course, Student will be able to:**
	1) Build Basic Database
	2) Build SQL statement
	3) Modify E-R model to relational table
	4) Construct integrity constraints

Module/Unit	Course Description	Hrs	CO No.	PSO No.	PO No.
1.	**Draw E-R diagram and convert into relation tables** a. Design E-R diagram b. Convert E-R diagram into relational database	3hrs	3	2	7
2.	**Design a Database and create required tables.** a. Creating College database b. Creating Bank database	3hrs	1		
3.	**Writing Basic SQL SELECT Statements** a. Restricting data b. Sorting Data	3hrs	1		
4.	**Applying the constraints** a. Table Level b. Column Level	3hrs	4		

5.	**Manipulating Data** a. Using INSERT b. Using UPDATE **c.** Using DELETE	3hrs	2		
6.	**Write a SQL statement for Creating and Managing Tables** a. Alter b. Drop	3hrs	2		
7.	**Write a queries using Group Functions and Single-Row Functions** a. SUM ( ) , AVG ( ),MIN ( ), MAX( ),COUNT() b. UPPER, LOWER and INITCAP.	3hrs	2		
8.	**Write the queries to implement the joins** a. Simple Join b. Outer Join	3hrs	2		
9.	**Write the queries to implement the set operators** a. UNION , UNION ALL b. INTERSECT c. MINUS	3hrs	4		
10.	**Write the query to create the database objects** a. Views b. Sequences	3hrs	1		
11.	**PL/SQL Basics** a. Declaring Variables b. Writing Executable Statements c. Interacting with the oracle server	3hrs	4		

References

1) Database Systems A Practical Approach To Design Implementation And Management 4th Edition Thomas Connolly Carolyn Begg,Person

Course Description: BSc(Information Technology)	
Semester	**II**
Course Name	**Numerical Methods Practical**
Course Code	**UIT2NMP**
Eligibility for the Course	
Credit	**2**
Hours	**3Hrs. per week**

Course Objectives	To familiarize the students with the fundamental concepts of scilab and develop programming skill to effectively implement for problems.

Course Outcomes	**After completing the course, Student will be able to:**
	1) Make use of the basic commands of scilab.
	2) Construct a formula for interpolation using scilab.
	3) Determine the differential equation and numerical integration using scilab.
	4) Design the program in scilab for Eigenvalue problems and linear systems of equations.

Module/ Unit	Course Description	Hrs.	CO No.	PSO No.	PO No.
1	**Solution of algebraic and transcendental equations** a. Program to solve algebraic and transcendental equations by bisection method. b. Program to solve algebraic and transcendental equation by false position method. c. Program to solve algebraic and transcendental equation by Secant method. d. Program to solve algebraic and transcendental equation by Newton Raphson method	3hrs	1	6	2
2.	**Solving linear system of equations by iterative methods** a. Program for solving linear systems of equations using Gauss Jordan method. b. Program for solving linear system of equations using Gauss Seidel method.	3hrs	1		

3.	**Interpolation I** a. Program for Newton's forward interpolation. b. Program for Newton's backward interpolation. c. Program for Newton's Divided Interpolation.	3hrs	2		
4.	**Interpolation II:** a. Program for Lagrange's interpolation. b. Program for Spline interpolation.	3hrs	2		
5.	**Numerical Differentiation** a. Programming to obtain derivatives numerically.	3hrs	3		
6.	**Solution of differential equations** a. Program to solve differential equation using Euler's method b. Program to solve differential equations using modified Euler's method. c. Program to solve differential equation using Runge-kutta 2nd order and 4th order methods	3hrs	3		
7.	**Numerical Integration** a. Program for numerical integration using Trapezoidal rule. b. Program for numerical integration using Simpson's 1/3rd rule. c. Program for numerical integration using Simpson's 3/8th rule.	3hrs	3		
8.	**Double Integration:** a. Program for numerical integration using Trapezoidal rule. b. Program for numerical integration using Simpson's rule.	3hrs	3		
9.	**Linear System of Equation:** a. Program for LU Decomposition b. Program for Partition Method	3hrs	4		
10.	**EigenValue Problem:** a. Program for Power Method	3hrs	4		

References:

1. Scilab textbook companion for numerical methods: principles, analysis and algorithms by S.Pal

Course Description: BSc(Information Technology)	
Semester	**II**
Course Name	**Web Programming Practical**
Course Code	**UIT2WPP**
Eligibility for the Course	
Credit	**2**
Hours	**3Hrs per week**

Course Objectives	To familiarize the students with basic web development concepts required to develop static web pages as well as advanced web concepts required for development of dynamic web pages.

Course Outcomes	**After completing the course, Student will be able to:**
	1) Create static web pages using HTML5 and CSS.
	2) Design a responsive website using HTML5 and CSS.
	3) Construct interactive web pages using javascript as client side scripting language.
	4) Develop dynamic web pages using PHP as server side scripting language.

Module/ Unit	**Course Description**	Hrs.	CO No.	PSO No.	PO No.
1.	**Use of Basic Tags** a. Design a web page using different text formatting tags. b. Design a web page with links to different pages and allow navigation between web pages. c. Design a web page demonstrating all Style sheet types	3hrs	1	2	7

2	**Image maps, Tables, Forms and Media** a. Design a web page with Image maps. b. Design a web page with different tables. c. Design a webpages using table so that the content appears well placed. d. Design a web page with a form that uses all types of controls. e. Design a web page using bootstrapping	3hrs	2		
3	**Java Script** a. 1. Using JavaScript design, a web page that prints factorial of a number. (Use prompts.) 2. Using JavaScript design, a web page that prints factorial of a number. (Use HTML form)3. Using JavaScript design, a web page that accepts number of terms from the user and displays Fibonacci series. (Use HTML form) b. Design a form and validate all the controls placed on the form using Java Script. c. Write a JavaScript program to accept a number from the user and display the sum of its digits. d. Write a JavaScript program to accept a number from the user and display whether it is prime or not. e. Write a program in JavaScript to accept a sentence from the user and	3hrs	3		

	display the number of words in it. (Do not use split () function). f. Write a java script program to design simple calculator.				
4	. **Control and looping statements and Java Script references** a. Design a web page 1. To accept a number from the user and display whether it is odd or even. (If…else). 2. To accept two numbers and operator from the user and perform the operation entered by user. (Use switch case) b. 1. Design a web page to display all the odd numbers from 1 to 50. (use while) 2. Design a web page to display all the numbers from 20 to 1. (use for) 3. Design a web page to demonstrate the use of for in loop on an array. c. Design a web page demonstrating different Core JavaScript references (Array, Date, Math, and String).	3hrs	3		
5	**Basic PHP I** a. Write a PHP Program to accept a number from the user and print it factorial. b. Write a PHP program to accept a number from the user and print whether it is prime or not.	3hrs	4		

<table>
<tr><td>6</td><td>Basic PHP II

a. Write a PHP code to find the greater of 2 numbers. Accept the no. from the user.

b. Write a PHP program to display the following Binary Pyramid:

<table><tr><td>1</td><td></td><td colspan="3"></td></tr><tr><td>0</td><td>1</td><td></td><td colspan="2"></td></tr><tr><td>1</td><td>0</td><td>1</td><td></td><td></td></tr><tr><td>0</td><td>1</td><td>0</td><td>1</td><td></td></tr></table></td><td>3hrs</td><td>4</td><td></td><td></td></tr>
<tr><td>7</td><td>String Functions and arrays

a. Write a PHP program to demonstrate different string functions.

b. Write a PHP program to create one dimensional array.</td><td>3hrs</td><td>4</td><td></td><td></td></tr>
<tr><td>8</td><td>PHP and Database

a. Write a PHP code to create:

● Create a database College

● Create a table Department (Dname, Dno, Number_Of_faculty)

b. Write a PHP program to create a database named “College”. Create a table named “Student” with following fields (sno, sname, percentage).

Insert 3 records of your choice. Display the names of the students whose percentage is between 35 to 75 in a tabular</td><td>3hrs</td><td>4</td><td></td><td></td></tr>
</table>

	format. c. Design a PHP page for authenticating a user.				
9	**Create a XML file with Internal/External DTD and display it using XSL**	3hrs			
10	**Sessions and Cookies** a. Write a program to demonstrate use of sessions and cookies.	3hrs	4		

References:

1) "HTML5 Step by Step", FaitheWempen , Microsoft Press
2) "JavaScript 2.0: The Complete Reference ", Thomas Powell and Fritz Schneider , Second Edition, Tata McGraw Hill Publication
3) "PHP 5.1 for Beginners", Ivan Bayross, Sharanam Shah, SPD Publication