UNIVERSITY OF MUMBAI

Syllabus for F.Y.B.Sc. Programme: B.Sc. Subject : Information Technology
Semester – I and II

(CBCS)
(Choice Based Credit System with effect from the academic year 2022-2023)
(To introduce with effect from the academic year 2022-2023)

UNIVERSITY OF MUMBAI

Syllabus for Approval

Sr. No.	Heading	Particulars
1	Title of the Programme	**F.Y.B.Sc. Sem. I & II (Information Technology)**
2	Eligibility for Admission	Ordinance no. O.5051 Circular no. UG/284 of 2007 dated 16th June 2007
3	Passing Marks	40%
4	Ordinances / Regulations (if any)	As applicable for all B.Sc. Courses
5	No. of Years / Semesters	Three years – Six Semesters
6	Level	~~P.G.~~ / U.G./ ~~Diploma~~ / ~~Certificate~~ **(Strike out which is not applicable)**
7	Pattern	~~Yearly~~ / Semester **(Strike out which is not applicable)**
8	Status	Revised / ~~New~~ /~~Amended~~ **(Strike out which is not applicable)**
9	To be implemented from Academic Year	From Academic Year **2022-2023**

Signature
Chairman Name Dr. R. Srivaramangai   Dr. Anuradha Majumdar
BOS Chairman in Information Technology   Dean, Science and Technology

PREAMBLE

The B.Sc. Information Technology programme was started in 2001 with an aim to make the students 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.

The new syllabus is aimed to achieve the objectives. The syllabus spanning three years covers the industry relevant courses. The students will be ready for the jobs available in different fields like:

- Software Development (Programming)
- Website Development
- Mobile app development
- Internet of Things
- Software Testing
- Networking
- Database Administration
- System Administration
- Cyber Law Consultant
- GIS (Geographic Information Systems)
- IT Service Desk
- Security
- Technical communication skills
- Green IT

And many others

Name of Dean (Dean, Faculty of Science and Technology) : Dr. Anuradha Majumdar

Name of Associate Dean (Associate Dean, Faculty of Science and Technology) Prof. Shivram Garje

   Name of Chairperson (BoS) : Dr. Mrs. R. Srivaramangai

   Member(BoS) : Dr. Hiren Dand

   Member(BoS) : Dr. Abhijeet Kale

   Member(BoS) : Dr. Santosh Singh

   Member(BoS) : Dr. Rajendra Patil

   Member(BoS) : Dr. Mandar Bhave

Semester 1			
Course Code	**Course Type**	**Course Title**	**Credits**
USIT101	Core Subject	Programming Principles with C	2
USIT102	Core Subject	Digital Logic and Applications	2
USIT103	Core Subject	Fundamentals of Database Management Systems	2
USIT104	Core Subject	Computational Logic and Discrete Structure	2
USIT105	Ability Enhancement Skill Course	Technical Communication Skills	2
USIT1P1	Core Subject Practical	Programming Principles with C Practical	2
USIT1P2	Core Subject Practical	Digital Logic and applications Practical	2
USIT1P3	Core Subject Practical	Fundamentals of Database Management Systems Practical	2
USIT1P4	Core Subject Practical	Computational Logic and Discrete structure Practical	2
USIT1P5	Ability Enhancement Skill Course Practical	Technical Communication Skills Practical	2
		Total Credits	**20**

Semester 2			
Course Code	**Course Type**	**Course Title**	**Credits**
USIT201	Core Subject	Object Oriented Programming with C++	2
USIT202	Core Subject	Fundamentals of Micro Processor and Microcontrollers	2
USIT203	Core Subject	Web Applications Development	2
USIT204	Core Subject	Numerical Methods	2
USIT205	Ability Enhancement Skill Course	Green IT	2
USIT2P1	Core Subject Practical	Object Oriented Programming with C++ Practical	2
USIT2P2	Core Subject Practical	Fundamentals of Micro Processor and Microcontrollers Practical	2
USIT2P3	Core Subject Practical	Web Applications Development Practical	2
USIT2P4	Core Subject Practical	Numerical Methods Practical	2
USIT2P5	Ability Enhancement Skill Course Practical	PL/SQL Practical	2
		Total Credits	**20**

SEMESTER I

B. Sc (Information Technology)		Semester – I	
Course Name: Programming Principles with C		Course Code: USIT101	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2	75
	Internal	--	25

Course Objectives: **1. To develop the logical ability of the student.**
2. Basic concepts to be cleared using suitable examples.
3. Different approach towards the problem.
4. To handle the errors and find suitable solution.
5. Debugging the code.

Unit	Details	Lectures
I	Introduction: Algorithms, History of C, Structure of C Program. Program Characteristics, Compiler, Linker and preprocessor, pseudo code statements and flowchart symbols, Desirable program characteristics. Program structure. Compilation and Execution of a Program, C Character Set, identifiers and keywords, data types and sizes , constants and its types, variables, Character and character strings, typedef, typecasting	12
II	Type of operators: Arithmetic operators, relational and logical operators, Increment and Decrement operators, assignment operators, the conditional operator, Assignment operators and expression, Precedence and order of Evaluation Block Structure, Initialization, C Preprocessor Control Flow: Statements and Blocks, If-Else, Else-If, Switch, Loops- While and For Loops- Do-while, Break and Continue, Goto and Labels	12
III	Functions and Program Structure: Basics of functions. User defined and Library functions, Function parameters, Return values, Recursion External variables, Scope Rules, Standard Input and Output, Formatted Output-printf() and Formatted Input- scanf(), Line Input and Output, Error Handling- StdErr and Exit, Header Files	12
IV	Pointer and Arrays Pointer and Addresses, Pointer and Function Arguments, Pointer and Arrays, Address Arithmetic, Character Pointers and Functions, Pointer Arrays: Pointers and Functions, Multidimensional Array, Command-line Arguments, Pointers to Functions, Dynamic memory allocation	12
V	Structures: Basics of structures, Structures and Functions, Arrays of Structures, Pointers to Structures, Unions, Bit-fields, File management in C: Defining and Opening file, Closing a file, Input / Output operations on file, Error handling in C, Random access to files, Command line arguments.	12

Books and References:					
Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Programming Language	Brian W. Kernighan and Denis M. Ritchie.	PHI	2nd	1988
2.	Mastering C	K R Venugopal	Tata McGraw-Hill	6th	2007
3.	Programming with C	Byron Gottfried	Tata McGRAW-Hill	2nd	1996
4.	Let us C	Yashwant P. Kanetkar	BPB publication		
5.	Programming in ANSI C	E.Balagurusamy	Tata McGraw-Hill	7th	1982

Course Outcomes:

Learners will be able to,

1. **Learn the basic principles of programming.**
2. **Develop of logic using algorithm and flowchart.**
3. **Acquire the information about data types.**
4. **Understanding of input and output functions.**
5. **Enhance advanced concepts using program.**

B. Sc (Information Technology)		Semester – I	
Course Name: Programming Principles with C Practical		Course Code: USIT1P1	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2½	50
	Internal	--	--

Course Objectives:
1. **To develop the logic of the student.**
2. **Describe loops and decision making using programs.**
3. **Practical use of operators.**
4. **Illustration of the difficult concepts using programming examples.**
5. **Discussion of the relevant concepts using program.**

List of Practical:
1.   a. Write an algorithm and draw flowchart for Area of circle. b. Write an algorithm and draw flowchart to print the given no. is even or odd. c. Write an algorithm and draw flowchart to print 1 to 10 numbers. d. Write an algorithm and draw flowchart for sum of 1 to 5 numbers. e. Write an algorithm and draw flowchart to compute the addition of digits of a given number.
2.   a. Write a program using while loop to reverse the digits of a number. b. Write a program to calculate the factorial of a given number. c. Write a program to find the roots of quadratic equation. d. Write a program to print the Fibonacci series.
3.   a. Write a program in C to check entered character vowel or consonant b. Write a program to C program to print day name of week using switch-case. c. Write a program to read three values from keyboard and print out the largest of them without using if statement.
4.   a. Write a program to print the pattern of asterisks as shown below : * * * * * * * * * * b. Write a program to print the pattern of asterisks as shown below : * * * * * * * * * * * * * * * **c. Write a program to print Floyd's Triangle.**
5.   a. Write a program to print area of square using function.

	b. Write a program using recursive function. **c. Write a program to square root, abs() value using function.** **d. Write a program using goto statement.**
6.	a. Write a program to print rollno and names of 10 students using array. b. Write a program to read a matrix of size m*n. **c. Write a program to sort the elements of array** in ascending or descending order.
7.	a. Write a program to extract the portion of a character string and print the extracted part. b. Write a program to find the given string is palindrome or not. **c. Write a program to using strlen(), strcmp() function.**
8.	a. Write a program to display the values using different data types and its address using pointer. b. Write a program to perform addition and subtraction using pointer.

9.	a. Write a program to copy the contents of the file from one file into other.
	b. Write a program to print the structure using
	    • Title
	    • Author
	    • Subject
	    • Book ID
	Print the details of two students.

10.	a. Create a mini project on "Bank management system" . The program should be menu driven.

Course Outcomes:

Learners will be able to,

1. **Develop applications.**
2. **Work with textual information, characters and strings.**
3. **Understand of a functional hierarchical code organization**
4. **Debug the program**
5. **Understand the differences between syntax errors, runtime errors, and logic errors.**

B. Sc (Information Technology)		Semester – I	
Course Name: Digital Logic and Applications		**Course Code: USIT102**	
Periods per week (1 Period is 50 minutes)		**5**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	**2**	**75**
	Internal	**--**	**25**

Course Objectives:
1. To introduce the basics of logic in digital electronics as an entry level course.
2. To interpret and assess number systems and the conversions of number systems
3. To analyze the boolean expressions and reduce the expression to the minimum.
4. To design simple logic circuits using tools such as Boolean Algebra and Karnaugh Mapping.
5. To understand the state of a memory cell and its types using flip-flops.
6. To create simple digital systems using counters, registers etc.

Unit	Details	Lectures
I	**Digital Systems and Binary numbers** Introduction to Number systems, Positional Number systems, Conversions (converting between bases), Non positional number systems, Unsigned and Signed binary numbers, Binary Codes, Number representation and storage in computer system. **Logic gates and Logic Circuits** Basic and Universal Gates	12
II	**Boolean algebra and Gate level minimization** Introduction, Postulates of Boolean Algebra, Two Valued Boolean Algebra, Principle of Duality, Basic Theorems of Boolean Algebra, Boolean Functions and their Representation, Gate-Level Minimization (Simplification of Boolean Function), Quine-McCluskey Method, Review questions	12
III	**Combinational logic** Introduction, Analysis and Design Procedure for Combinational Logic Circuits, Types of Combinational Circuit, Review Questions	12
IV	**Sequential circuits** Introduction, Latch, Flip-Flops, Registers, Counters, Review Questions	12
V	**Applications** Bit Arithmetic and Logic unit, Carry lookahead generator, Binary Multiplication and Division algorithm, Booth's multiplication algorithm	12

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Digital Logic Design	Sonali Singh	BPB publications	1st	2015
2.	Fundamentals of Digital Electronics and Logic Design	Subir Kumar Sarkar, Asish Kumar De, Souvil Sarkar	Pan Stanford Publishing	1st	2014
3.	Digital Electronics Principles, Design and Applications	Anil K Maini	Wiley	1st	2007
4.	Fundamentals of Logic	Charles H Roth, Jr. , Larry L	Cengage	7th	2014

	Design	Kinney	Learning		
5.	Digital Principles and Applications	Donald P Leach Albert Malvino Goutam Saha	TMH	8th	2015

Course Outcomes:

Learners will be able to,

1. Apply number conversion techniques in real digital systems
2. Solve boolean algebra expressions
3. Derive and design logic circuits by applying minimization in SOP and POS forms
4. Design and develop Combinational and Sequential circuits
5. Understand and develop digital applications

B. Sc (Information Technology)		Semester – I	
Course Name: Digital Logic and Applications Practical		**Course Code: USIT1P2**	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	**Marks**
Evaluation System	**Practical Examination**	2½	50
	Internal	--	--

Course Objectives:
1. To apply and test the gates learnt using various IC's .
2. To evaluate the Boolean expression to reduce and minimize the gates used

1.	**Study of basic gates and Universal gates**
a.	To verify the truth tables of OR, AND, NOR, NAND, EX-OR, EX-NOR gates
b.	To study IC 7400, 7402, 7404, 7408, 7432, 7486, 74266
c.	To implement and verify NAND and NOR as Universal gates
2.	**Study of Boolean expressions**
a.	To verify De Morgan's laws
b.	Implement the given expression using a minimum number of gates.
c.	Implement the given expression using a minimum number of ICs.
3.	**Design of Combinational Circuits using K-maps**
a.	Design and implement combinational circuits for the given problem/problems using minimization techniques of K-maps.
4.	**Design and implement code converters**
a.	Design the circuit and implement Binary to gray code converter
b.	Design the circuit and implement Gray to Binary code converter
c.	Design the circuit and implement Binary to BCD code converter
d.	Design the circuit and implement Binary to XS-3 code converter
5.	**Implement Adder and Subtractor circuits**
a.	Design the circuit and implement Half Adder and Full Adder
b.	Design the circuit and implement BCD Adder, XS-3 Adder , Binary Subtractor

6.	**Design and implement Arithmetic circuits**
a.	Design and implement 2-by-2 bit multiplier
7.	**Implement Encoders and Decoders**
a.	Design and implement 8: 3 encoder
b.	Design and implement 3:8 decoder
8.	**Multiplexers and Demultiplexers**
a.	Design and Implement 4:1 multiplexer
b.	Design and Implement 1:4 demultiplexer
c.	Study IC 74151 8: 1 multiplexer and implement the expression
d.	Study IC 74138 3: 8 decoder and implement the expression
9.	**Study of Flipflops and Counters**
a.	Study of IC's 7473, 7474, and 7476
b.	Design a 3-bit ripple/ synchronous counter using IC 7473 and required gates
10.	**Design of Shift Registers**
a.	Design of Shift registers using IC 7474
b.	Implementation of digits using seven segment displays

Course Outcomes:

Learners will be able to,

1. **Construct basic and universal logic circuits.**
2. **Verify the functionalities of various IC's.**
3. **Design circuits using K-maps minimization technique**
4. **Design and test Encoders, Decoders, Multiplexers and Demultiplexers**
5. **Design and develop logic for Registers, Counters and its applications.**

B. Sc (Information Technology)		Semester – I	
Course Name: Fundamentals of Database Management Systems		Course Code: USIT103	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2	75
	Internal	--	25

Course Objectives:

The objective of the course is to present an introduction to fundamentals of database management systems, with an emphasis on how to organize, maintain and retrieve - efficiently, and effectively - information from a DBMS.

Unit	Details	Lectures
I	Database system- concept and Architecture, Relational model and Relational database constraints. Relational Algebra.	12
II	Conceptual modelling and database design: Data modelling using the Entity Relationship model (ER).The enhanced entity relationship model. Relational database design by ER and EER model. Practical database design methodology and use of UML diagrams.	12
III	Database Design theory and normalization: Basics of functional dependencies and normalization for relational databases. Relational database design and further dependencies.	12
IV	Introduction to SQL , Complex queries, triggers, views, joining database tables and schema modification. Query Processing and optimization. File structure, hashing and indexing	12
V	Transaction management and concurrency control and recovery: Introduction to transaction processing concepts and theory. Concurrency control technique. Database recovery technique.	12

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	**Fundamentals of Database systems.**	**Ramez Elmasri, Shamkant B Navathe**	Pearson.	**6th Edition**	
2.	**Database Systems: Design implementation and management.**	**Carlos Coronel, Steven Morris, Peter Rob**	Cengage Learning	**9th Edition**	2010

Course Outcomes:

Learners will be able to

1. **Define and describe the fundamental elements of relational database management system.**
2. **To relate the basic concepts of relational data model, entity-relationship model, relational database design, relational algebra and SQL.**
3. **Design ER-models to represent simple database application scenarios.**
4. **Transform the ER-model to relational tables, populate relational database and formulate SQL queries on data.**
5. **Improve the database design by normalization.**
6. **Understand basic database storage structures and access techniques: file and page organizations, indexing methods and hashing.**

B. Sc (Information Technology)		Semester – I	
Course Name: Fundamentals of Database Management Systems Practical		Course Code: USIT1P3	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2½	50
	Internal	--	--

Course Objectives:
- **To introduce ER data model, database design and normalization.**
- **To Learn SQL basics for data definition and data manipulation.**

1.	Draw E-R diagram and convert entities and relationships to relation table for a given scenario
a.	Bank
b.	College
2.	**Write relational algebra queries for a given set of relations**
3.	**Defining data**
a.	Using CREATE statement
b.	Using ALTER statement
c.	Using DROP statement
d.	Using TRUNCATE statement
e.	Using RENAME statement
4.	**Manipulating data**
a.	Using INSERT statement
b.	Using UPDATE statement
c.	Using DELETE statement
d.	Using SELECT statement
5.	**Creating and managing the tables**
a.	Creating table with contraints: NOTNULL, UNIQUE, PRIMARY KEY ,FOREIGN KEY
6.	**Restricting and sorting data**
a.	Using DISTINCT,IN, AS, SORT,LIKE,ISNULL, OR
b.	Using Group By, Having clause, Order By clause
7.	**Aggregate and Mathematical functions:**
a.	AVG,MIN,MAX,SUM,COUNT

b.	ABS,SQRT,ROUND,TRUNCATE,SIGN,POWER,MOD,FLOOR,CEIL
8.	**Views and Joins: For a given set of relation tables perform the following**
a.	Creating view
b.	Dropping view
c.	Selecting from a view
8.	**Database trigger**
a.	Using CREATE OR REPLACE TRIGGER
9.	**Index**
a.	Create index
b.	Drop index

Course Outcomes:

Learners will be able to:

1. **Design database schema for a given application and apply normalization.**
2. **Acquire skills in using SQL Commands for data Definition and data manipulation.**

B. Sc (Information Technology)		Semester – I	
Course Name: Computational Logic and Discrete Structures		Course Code: USIT104	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2	75
	Internal	--	25

Course Objectives:

- **Course will provide students with an overview of discrete mathematics.**
- **Students will learn about topics such as logic and proofs, sets and functions, recursion, graph theory, tress and other important discrete math concepts.**

Unit	Details	Lectures
I	**Set Theory** Introduction, Sets and Elements, Subsets, Venn Diagrams, Set Operations, Algebra of Sets, Duality, Finite Sets, Counting Principle, Classes of Sets, Power Sets, Partitions, Mathematical Induction **Relations** Introduction, Product Sets, Relations, Pictorial Representatives of Relations, Composition of Relations, Types of Relations, Closure Properties, Equivalence Relations, Partial Ordering Relations	12
II	**Functions and Algorithms** Introduction, Functions, One-to-One, Onto, and Invertible Functions, Mathematical Functions, Exponential and Logarithmic Functions, Sequences, Indexed Classes of Sets, Recursively Defined Functions, Cardinality, Algorithms and Functions, Complexity of Algorithms **Probability** Introduction, Sample Space and Events, Finite Probability Spaces, Conditional Probability, Independent Events, Independent Repeated Trials, Binomial Distribution, Random Variables, Chebyshev's Inequality, Law of Large Numbers	12
III	**Techniques of Counting** Introduction, Basic Counting Principles, Mathematical Functions, Permutations, Combinations, the Pigeonhole Principle, The Inclusion–Exclusion Principle, Tree Diagrams **Advanced Counting Techniques, Recursion** Introduction, Combinations with Repetitions, Ordered and Unordered Partitions, Inclusion–Exclusion Principle Revisited, Pigeonhole Principle Revisited, Recurrence Relations, Linear Recurrence Relations with Constant Coefficients, Solving Second-Order Homogeneous Linear Recurrence, Relations, Solving General Homogeneous Linear Recurrence Relations	12
IV	**Graph Theory** Introduction, Data Structures,,Graphs and Multigraphs, Subgraphs, Isomorphic and Homeomorphic Graphs, Paths, Connectivity, Traversable and Eulerian Graphs, Bridges of Königsberg, Labeled andWeighted Graphs, Complete, Regular, and Bipartite	12

	Graphs, Tree Graphs, Planar Graphs, Graph Colorings, Representing Graphs in Computer Memory, Graph Algorithms, Traveling-Salesman Problem, Solved Problems **Directed Graphs** Introduction, Directed Graphs, Basic Definitions, Rooted Trees, Sequential Representation of Directed Graphs, Warshall's Algorithm, Shortest Paths, Linked Representation of Directed Graphs, Graph Algorithms: Depth-First and Breadth-First Searches, Directed Cycle-Free Graphs, Topological Sort, Pruning Algorithm for Shortest Path	
V	**Binary Trees** Introduction, Binary Trees,, Complete and Extended Binary Trees, Representing Binary Trees in Memory, Traversing Binary Trees, Binary Search Trees, Priority Queues, Heaps, Path Lengths, Huffman's Algorithm, General (Ordered Rooted) Trees Revisited **Ordered Sets and Lattices** Introduction, Ordered Sets, Hasse Diagrams of Partially Ordered Sets, Consistent Enumeration, Supremum and Infimum, Isomorphic (Similar) Ordered Sets, Well-Ordered Sets, Lattices 346 Bounded Lattices, Distributive Lattices, Complements, Complemented Lattices	12

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Discrete Mathematics, Schaum's Outlines Series	Seymour Lipschutz, Marc Lipson	Tata MCGraw Hill	3rd	2007
2.	Discrete Mathematics with Applications	Sussana S. Epp	Cengage Learning	5th	2018
3.	Discrete Mathematics and its Applications	Kenneth H. Rosen	Tata MCGraw Hill	8th	2019
4.	Discrete mathematical structures	B Kolman RC Busby, S Ross	PHI		
5.	Discrete structures	Liu	Tata MCGraw Hill		

Course Outcomes:

Learners will be able to:

1. **Use logical notation**
2. **Perform logical proofs**
3. **Apply recursive functions and solve recurrence relations**
4. **Use graphs and trees**
5. **Apply basic and advanced principles of counting**
6. **Define sets and Relations**
7. **Calculate discrete probabilities.**

B. Sc (Information Technology)	Semester – I
Course Name: Computational Logic and Discrete Structures Practical	**Course Code: USIT1P4**
Periods per week (1 Period is 50 minutes)	3
Credits	2

		Hours	Marks
Evaluation System	**Practical Examination**	2½	50
	Internal	--	--

Course Objectives:

Course will make students understand different commands and functions of SCILAB. It will enable student to use these tools to compute solutions of various discrete mathematical structures.

1.	**Set Theory**
a.	Inclusion Exclusion principle.
b.	Power Sets
c.	Mathematical Induction
2.	**Functions and Algorithms**
a.	Recursively defined functions
b.	Cardinality
c.	Polynomial evaluation
d.	Greatest Common Divisor
3.	**Probability Theory 1**
a.	Sample space and events
b.	Finite probability spaces
c.	Equiprobable spaces
d.	Addition Principle
4.	**Probability Theory 2**
a.	Conditional Probability
b.	Multiplication theorem for conditional probability
c.	Independent events
d.	Repeated trials with two outcomes
5.	**Counting 1**
a.	Sum rule principle
b.	Product rule principle
c.	Factorial
d.	Binomial coefficients
6.	**Counting 2**
a.	Permutations
b.	Permutations with repetitions

c.	Combinations
d.	Combinations with repetitions
7.	**Counting 3**
a.	Ordered partitions
b.	Unordered partitions
8.	**Graph Theory**
a.	Paths and connectivity
b.	Minimum spanning tree
c.	Isomorphism
9.	**Directed Graphs**
a.	Adjacency matrix
b.	Path matrix
10	**Recurrence relations**
a.	Linear homogeneous recurrence relations with constant coefficients
b.	Solving linear homogeneous recurrence relations with constant coefficients
c.	Solving general homogeneous linear recurrence relations

Course Outcomes:

Learners will be able to:

1. **To find computational solution to various discrete mathematical structures.**

B. Sc (Information Technology)		Semester – I	
Course Name: Technical Communication Skills		**Course Code: USIT105**	
Periods per week (1 Period is 50 minutes)		**5**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	2	75
	Internal	--	25

Course Objectives:
- **To recognize the importance of various types of communication in technical set up.**
- **To understand the dynamics in different forms of formal communication.**
- **To learn about active listening and the art of giving presentations and interviews.**
- **To learn the art of business writing and ethics in business communication across functional areas.**
- **To evaluate, analyze and interpret technical data.**

Unit	Details	Lectures
I	**Fundamentals of Technical Communication** Introduction, The process of communication, Language as tool of communication, levels of communication, The flow of communication, Communication Networks, The importance of technical communication **Barriers to communication** Definition of Noise, classification of Barriers **Non-verbal Communication** Introduction, Definition, significance of nonverbal, forms of non -verbal communication, types of non-verbal communication	**12**
II	**The Seven Cs of Effective Communication**: Completeness, Conciseness, Consideration, Concreteness, Clarity, Courtesy, Correctness **Conversations** Introduction, Importance of Business conversion, Essential of Business conversion, Conversation Management **Meeting and conferences** Introduction, Purpose of Meeting, planning a meeting, Meeting Process, Leading effective meeting, Evaluating meeting, planning conference, teleconferencing **Group Discussion and team presentation** Introduction, Benefits of GD, Workplace GD guidelines, Functional and non functional roles in GD, Improving group performance, Assessment of group discussion ,Team presentation **Email communication** Introduction, Advantages of email, problems in email communication, Email etiquettes, Techniques of writing Effective Email	**12**
III	**Active Listening** Introduction, Type of listening, Traits of good listener, Active vs Passive listening, Implication of effective listening **Effective presentation Strategies** Introduction, Defining purpose, Analyzing audience and Locale, Organizing contents, preparing outline, Visual Aids, Understanding Nuances of delivery, Kinesics **Interview** Introduction, objectives, types of interview, job interviews	**12**
IV	**Business writing** Introduction, Importance of written Business, Five main strategies of writing business messages	**12**

	Business correspondence Business letter writing, common component of Business letter, Strategies for writing body of a letter, Types of Business letter, writing memos **Business reports and proposal** What is report? Steps in writing routine Business report, parts of report, corporate reports and Business proposals **Careers and Resume** Introduction to career building, resume format, traditional, electronic and video resumes, sending resume, follow up letters and online recruitment process	
V	**Communication across Functional areas** Financial communication, MIS **Ethics in Business Communication** Ethical communication, Values, ethics and communication, ethical dilemmas facing manager, strategic approaches to corporate ethics **Creating and Using Visual Aids** Object, Models, Handouts, Charts and Graphs, Text Visuals , Formatting Computer generated charts, graphs and visuals	**12**

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Technical communication : principles and practices	Meenakshi Raman & Sangeeta Sharma	Oxford Higher Education		
2.	Business Communication	Meenakshi Raman & Prakash Singh	Oxford- Higher Education	2nd edition	2006
3.	Effective Business Communication	Herta Murphy, Herbert Hildebrandt, Jane Thomas	Tata McGraw Hill	7th edition	2008
4.	Professional Communication	Aruna Koneru	McGraw Hill		2008
5.	Business and Professional Communication Plans, Processes and Performance	James R. DiSanza Nancy J..Legge	Pearson Education	4th Edition	
6.	Storytelling with data-a data visualization guide for business professionals	Cole Nussbaumer knaflic	Wiley		

Course Outcome:

Learners will be able to,

1. **Analyze, synthesize and utilize the process and strategies from delivery to solving communication problem.**
2. **Learn the communication methodologies at workplace and learning about importance of team collaboration.**
3. **Learn about different technical communication such as presentations and interviews.**
4. **Understand and apply the art of written communication in writing reports, proposals.**
5. **Ground rules of ethical communication and MIS.**
6. **Understand the functions of graphs, maps, charts.**

B. Sc (Information Technology)		Semester – I	
Course Name: Technical Communication Skills Practical		**Course Code: USIT1P5**	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	**Marks**
Evaluation System	**Practical Examination**	2½	50
	Internal	--	--

Course Objectives:
- **To express thoughts feelings and ideas of learners by using features of MS Word.**
- **To articulate formal and informal reports.**
- **To analyze and interpret data and learn visualization of data.**
- **To learn effective tools of presentation**.

1.	Use of word processing tools for communication.
a.	Use of various tools like spell checker, header, footer etc.
b.	Make formal and informal letters, creating resume.
c.	Designing brochures and flyers using templates in word.
2.	Writing reports, minutes of meeting, action plan.
3.	Use of spreadsheet for data interpretation and data analysis.
4.	Basic use of what if analysis using excel.
5.	Visual Representation of data using excel – pie chart ,line chart, bar chart etc.
6.	Summarization of data using of pivot tables and chart in excel.
7.	Use of presentation tools like PowerPoint for communication and presentation skills.
8.a.	Basic communication covering the following topics:- Meeting people, Asking Questions and Design of questionnaire.
8.b.	Using netiquettes in online mode of communication using Zoom / Google Meet / MS-Teams etc.
9.	Use of Mail etiquette for writing effective mails.
10.a.	Use of Mail merge and its features.
10.b.	Creating profile using Linkedln.

Course Outcome:

Learners will be able to:

1. **Use different forms of digital mediums for effective communication.**
2. **Create technical documents and format existing documents for effective communication.**
3. **Learn to use graphical tools for better visualization.**
4. **Create business presentation effectively.**
5. **Visualize the data from pictorial representations**.

SEMESTER II

B. Sc (Information Technology)		Semester – I
Course Name: Object Oriented Programming with C++		**Course Code: USIT201**
Periods per week (1 Period is 50 minutes)		**5**
Credits		**2**
		Hours	**Marks**
Evaluation System	**Theory Examination**	2	75
	Internal	--	25

Course Objectives:

Understand object oriented programming and advanced C++ concepts

- **Be able to explain the difference between object oriented programming and procedural programming.**
- **Be able to program using more advanced C++ features such as composition of objects, operator overloads, dynamic memory allocation, inheritance and polymorphism, file I/O, exception handling, etc.**
- **Be able to build C++ classes using appropriate encapsulation and design principles**
- **Be able to apply object oriented or non-object oriented techniques to solve**
- **bigger computing problems**

Unit	Details	Lectures
I	**INTRODUCTION OF OBJECT-ORIENTED DESIGN** : Introduction, Objects, Class and Instance, Polymorphism, Inheritance, Object-Oriented Analysis , Finding the Objects ,Conceptual Modeling Requirements Model , Analysis Model, The Design Model, The Implementation Model , Test Model, Object-Oriented Analysis and Design, The Evolution of Object Model, Object-Oriented Programming, Object-Oriented Design, Object-Oriented Analysis, Elements of Object Model ,The Role of OOAD in the Software Life Cycle, OOAD Methodologies, Grady Booch Approach, **STARTING WITH C++:** C++ Overview, C++ Character Set, C++ Tokens, Variables, Counting Tokens, Data Types, Qualifiers, Range of Data Types, Your First C++ Program, Structure of a C++ Program, Styles of, Writing C++ Programs, Programming Examples **FEATURES OF C++:** Introduction, Operators and Expressions, Declaring Constants, Type Conversion, Decision Making: An Introduction, Unconditional Branching Using Goto , Introduction to Looping **OPERATORS AND REFERENCES IN C++**: Introduction, Scope Resolution Operator, Reference Variables, The Bool Data Type, The Operator New and Delete, Malloc Vs. New ,Pointer Member Operators	**12**
II	**FUNCTION IN C++**: Introduction ,Function Declaration/Prototyping ,,The Main Function in C++,Recursion ,Call by Reference ,Call by Reference Vs Call by Address , Return by Reference ,,Inline Function ,Function Overloading ,Function with Default Arguments **CLASS AND OBJECTS IN C++** : Working with Class, Structure in C++ ,Accessing Private Data Passing and Returning Object ,Array of Object ,Friend Function ,Static Class Members ,Constant Member Function **WORKING WITH CONSTRUCTOR AND DESTRUCTOR:** Introduction, Constructor with Parameters, Implicit and Explicit Call to Constructor, Copy Constructor, Dynamic Initialization of Objects, Dynamic Constructor, Destructor **WORKING WITH OPERATOR OVERLOADING:** Introduction, Operator Overloading with Binary Operator Overloading Assignment (=) Operator, Overloading	**12**

	Unary Operators, Overloading Using Friend Function, Rules of Operator Overloading, Type Conversion	
III	**WORKING WITH INHERITANCE IN C++:** Introduction, Types of Inheritance, Public, Private and Protected Inheritance, Multiple Inheritance, Hierarchical Inheritance, Virtual Base Class, Constructor and Destructor in Inheritance, Containership **POINTERS TO OBJECTS AND VIRTUAL FUNCTIONS:** Pointer to Objects, The This Pointer, What is Binding in C++? , Virtual Functions ,Working of a Virtual Function ,Rules for Virtual Function ,Pure Virtual Function and Abstract Class ,Object Slicing ,Some Facts about Virtual Function ,Virtual Destructor **INPUT-OUTPUT AND MANIPULATORS IN C++:** Introduction, C++ Stream Classes, Unformatted Input/Output, Formatted Input /Output Operations, Manipulators	**12**
IV	**FILE HANDLING IN C++:** Introduction, File Streams, Opening and Closing a File, File Opening Modes Checking End of File, Random Access in File, Command Line Arguments, Working with Binary Mode Error Handling **TEMPLATE PROGRAMMING:** Introduction , Function Template , Class Template **EXCEPTION HANDLING IN C++ :** Introduction , Basics of Exception Handling , Exception Handling Mechanism , Programming Examples ,Exception Handling with Class Catching all Exceptions , Specifying Exception for a Function	**12**
V	**INTRODUCTION TO THE STANDARD TEMPLATE LIBRARY** : Introduction , Components of STL , Containers , Algorithms , Iterators ,Application of Container Classes Function Objects **MANIPULATING STRINGS** :Introduction , Creating (string) Objects , Manipulating String Objects , Relational Operations, String Characteristics , Accessing Characters in Strings , Comparing and Swapping **NEW FEATURES OF ANSI C++ STANDARD** : Introduction ,New Data Types , New Operators , Class Implementation , Namespace Scope , Operator Keywords , New Keywords , New Headers	**12**

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Object-oriented Programming C++ Simplified	Hari Mohan Pandey	University Science Press	1st Edition	2017
2.	Object Oriented Programming in C++	E Balagurusamy	Tata McGraw-Hill	5th Edition	2011
3.	Object-Oriented Programming in C++	Robert Lafore	Sams	4th Edition	2002
4.	Programming with ANSI C++	Bhushan Trivedi	Oxford University Press	2nd Edition	2012
5.	Demystified Object-Oriented Programming with C++	Dorothy R. Kirk	Packt Publishing Lt	1st Edition	2021
6.	C++ Programming: An Object-Oriented Approach	Behrouz A. Forouzan , Richard F. Gilberg	McGraw-Hill Education	1st edition	2020
7.	C++ How to Program	Paul Deitel, Harvey Deitel	Pearson Education	10th Edition	2017

Annexure I

Course Outcomes:

Learners will be able to,

1. Understand the concept of OOPs, feature of C++ language.
2. Understand and apply various types of Datatypes, Operators, Conversions while designing the program.
3. Understand and apply the concepts of Classes &Objects, friend function, constructors & destructors in program design.
4. Design & implement various forms of inheritance, String class, calling base class constructors.
5. Apply & Analyze operator overloading, runtime polymorphism, Generic Programming.
6. Analyze and explore various Stream classes, I/O operations and exception handling.

B. Sc (Information Technology)		Semester – II	
Course Name: Object Oriented Programming with C++ Practical		Course Code: USIT2P1	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	**Marks**
Evaluation System	Practical Examination	2½	50
	Internal	--	--

Course Objectives:

- The student should be able to explain the important characteristics of the C++ programming language.
- The learner must be able to combine components of the C++ programming language to develop structured program.
- The student must demonstrate the skills essential to compile, debug, and test C++ programs correctly.

1.	
a.	Write a C++ program to create a simple calculator.
b.	Write a C++ program to convert seconds into hours, minutes and seconds.
c.	Write a C++ program to find the volume of a square, cone, and rectangle.
2.	
a.	Write a C++ program to find the greatest of three numbers.
b.	Write a C++ program to find the sum of even and odd n natural numbers
c.	Write a C++ program to generate all the prime numbers between 1 and n, where n is a value supplied by the user.
3.	
a.	Write a C++ program using classes and object Student to print name of the student, roll_no. Display the same.
b.	Write a C++ program for Structure bank employee to print name of the employee, account_no. & balance. Display the same also display the balance after withdraw and deposit
c.	Write a C++ Program to design a class having static member function named showcount() which has the property of displaying the number of objects created of the class.

d.	Write a Program to find Maximum out of Two Numbers using friend function. Note: Here one number is a member of one class and the other number is member of some other class.
e.	Write a C++ Program using copy constructor to copy data of an object to another object.
f.	Write a C++ Program to allocate memory dynamically for an object of a given class using class's constructor.
4.	
a.	Write a C++ program to design a class representing complex numbers and having the functionality of performing addition & multiplication of two complex numbers using operator overloading.
b.	Write a C++ program to overload new/delete operators in a class.
c.	Write a C++ program to access members of a STUDENT class using pointer to object members
	Write a C++ Program to generate Fibonacci Series by using Constructor to initialize the Data Members.
d.	Write a C++ Program to generate Fibonacci Series by using Constructor to initialize the Data Members.
e.	Write a C++ Program that illustrate single inheritance.
f.	Write a C++ Program that illustrate multipe inheritance.
g.	Write a C++ Program that illustrate multi level inheritance.
h.	Write a C++ Program that illustrate Hierarchical inheritance.
i.	Write a C++ Program illustrating how the constructors are implemented and the order in which they are called when the classes are inherited. Use three classes named alpha, beta, gamma such that alpha,beta are base class and gamma is derived class inheriting alpha & beta
5.	
a.	Write a C++ Program to design a stuent class representing student roll no. and a test class (derived class of student) representing the scores of the student in various subjects and sports class representing the score in sports. The sports and test class should be inherited by a result class having the functionality to add the scores and display the final result for a student.
6.	
a.	Write a C++ program to maintain the records of person with details (Name and Age) and find the eldest among them. The program must use this pointer to return the result.
7.	
a.	Write a C++ program illustrating the use of virtual functions in class.
b.	Write a C++ program to design a class representing the information regarding digital library (books, tape: book & tape should be separate classes having the base class as media). The class should have the functionality for adding new item, issuing, deposit etc. the program should use the runtime polymorphism.
8.	
a.	Write a C++ program to show conversion from string to int and vice-versa.
b.	Write a C++ program implementing basic operation of class ios i.e. setf, unsetf, precision etc.
c.	Write a C++ program to implement I/O operations on characters. I/O operations includes inputting a string, Calculating length of the string, Storing the String in a file, fetching the stored characters from it, etc.
d.	Write a C++ program to copy the contents of one file to another.

e.	Write a C++ program to perform read/write binary I/O operation on a file (i.e. write the object of a structure/class to file).
9.	
a.	Write a C++ program to implement the exception handling with multiple catch statements.
b.	Write a C++ program to implement the exception handling with rethrowing in Exception.
10.	
a.	Write a C++ Program to create Simple calculator using Class template.
b.	Write a C++ Program to get maximum of two number using Class template.

Course Outcomes:

Leaners will able to,

- **Utilize C++ characteristics in software design and development.**
- **Explain object-oriented techniques and explain how C++ supports them.**
- **Employ C++ to demonstrate practical skill developing object-oriented solutions.**
- **Examine a problem statements and design and develop object-oriented software using good coding practices and procedures.**
- **In object-oriented design, use common software patterns and recognize their relevance in other software development contexts.**

B. Sc (Information Technology)		Semester – II	
Course Name: Fundamentals of Micro Processor and Microcontrollers		**Course Code: USIT202**	
Periods per week (1 Period is 50 minutes)		**5**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	2	75
	Internal	--	25

Course Objectives: 1) To understand the basic concept of Micro Computer Systems
2) To develop background knowledge in 8085 Microprocessor
3) To write Assembly language Programs of 8085
4) To understand the peripheral devices and interfacing to 8051 Micro
Controller and design aspects of Micro Controller

Unit	Details	Lectures
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 operation's, Memory, I/O Devices, Microcomputer System, Logic Devices and Interfacing, Microprocessor-Based System Application. **8085 Microprocessor Architecture and Memory Interface:** Introduction, 8085 Microprocessor unit, 8085-Based Microcomputer, Memory Interfacing,Interfacing the8085 Memory Segment.	12
II	**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. **Introduction to 8085 Assembly Language Programming:** The 8085 Programming Model, Instruction Classification, Instruction, Data and Storage, Writing assembling and Execution of a simple program, Overview of 8085 Instruction Set, Writing and Assembling Program. **Introduction to 8085 Instructions:** Data Transfer Operations, Arithmetic Operations, Logic Operation, Branch Operation, Writing Assembly Languages Programs, Debugging a Program.	12
III	**Programming Techniques With Additional Instructions:** ProgrammingTechniques: 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. **Stacks and Sub-Routines:** Stack, Subroutine, Restart, Conditional Call, Return Instructions, Advanced Subroutine concepts. **Interrupts:** The 8085 Interrupt, 8085 Vectored and Non vectored Interrupts, Restart as S/W Instructions.	12

IV	**Micro Controllers:** Embedded Systems and general purpose computer systems, history, classifications, applications and purpose of embedded systems. **Embedded Hardware:** Memory map, i/o map, interrupt map, processor family, external peripherals, memory – RAM , ROM, types of RAM and ROM, memory testing, CRC ,Flash memory. **Peripherals:** Control and Status Registers, Device Driver, Timer watch Timer **The 8051 Microcontrollers:** Microcontrollers and Embedded processors, Overview of 8051 family.8051 Microcontroller hardware, Input/output pins, Ports, and Circuits, External Memory. **8051 Programming in C:** Data Types and time delay in 8051 C, I/O Programming, Logic operations, Data conversion Programs.	**12**
V	**Designing Embedded System with 8051 Microcontroller:**Factors to be considered in selecting a controller, why 8051 Microcontroller, Designing with 8051. **Programming embedded systems:** structure of embedded program, infinite loop, compiling, linking and debugging. **Design and Development:** Embedded system, development Environment – IDE, types of file generated on cross compilation, Embedded Product Development cycle and Trends in embedded Industry	**12**

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Microprocessors Architecture, Programming and Applications with the 8085.	Ramesh Gaonkar	PENRAM	5th	2012
2.	8080A/8085 Assembly Language Programming	Lance A. Leventhel	Osborne		1978
3	Embedded Systems	Rajkamal	Tata Mcgraw-Hill		
4	Introduction to embedded systems	Shibu K V	Tata Mcgraw-Hill	1st	2012

Course Outcomes:

Learners will be able to,

1. **Understand the basic concepts of Micro Computer Systems**
2. **Understand the architecture and hardware aspects of 8085**
3. **Write assembly language programs in 8085**
4. **Design elementary aspects of Micro Controller based systems**
5. **Interfacing peripherals using Micro Controller**

B. Sc (Information Technology)		Semester – II	
Course Name: Fundamentals of Micro Processor and Microcontrollers Practical		Course Code: USIT2P2	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2½	50
	Internal	--	--

Course Objectives: The course aims to learn
1. Operations related to single & Multiple memory locations
2. Simple assembly language programs
3. How to perform register operations, packing and unpacking
4. Embedding computer using 8051 microcontrollers
5. Interfacing I/O Ports
6. Implement flashmagic in microcontrollers

List of Practical	
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
2.	**Simple assembly language programs.**
a.	Subtract two 8-bit numbers.
b.	Subtract the 16-bit number in memory locations 4002H and 4003H from the 16-bit number in memory locations 4000H and 4001H. The most significant eight bits of the two numbers 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.
c.	Find the l's complement of the number stored at memory location 4400H and store the complemented number at memory location 4300H.
d.	Find the 2's complement of the number stored at memory location 4200H and store the complemented number at memory location 4300H.
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.
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.	
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. b. 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.	Divide 16 bit number stored in memory locations 2200H and 2201H by the 8 bit number stored at memory location 2202H. Store the quotient in memory locations 2300H and 2301H and remainder in memory locations 2302H and 2303H.	
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 memory location 2Sample problem:	
7.	**Assembly programs on memory locations**	
a.	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	
b.	Write an assembly language program to generate fibonacci number.	
c.	Program to calculate the factorial of a number between 0 to 8.	
8.	Design and develop a reprogrammable embedded computer using 8051 microcontrollers and to show the following aspects. a. Programming b. Execution c. Debugging	
9 a	Configure timer control registers of 8051 and develop a program to generate given time delay.	
b	Port I / O: Use one of the four ports of 8051 for O/P interfaced to eight LED's. Simulate binary counter (8 bit) on LED's	
c	To interface 8 LEDs at Input-output port and create different patterns.	
d	To demonstrate timer working in timer mode and blink LED without using any	

		loop delay routine.
10.		**Using FlashMagic**
	a.	To demonstrate the procedure for flash programming for reprogrammable embedded system board using Flash Magic
	b.	To demonstrate the procedure and connections for multiple controllers programming of same type of controller with same source code in one go, using flash magic.

Course Outcomes:

Learners will be able to,

1. **Apply concepts of 8085 to single & Multiple Memory Locations**
2. **Apply concepts of micro-processor register operations**
3. **Can implement assembly language programs**
4. **Use of Shift registers 8 & 16 bits**
5. **Apply the knowledge of Flash Magic in embedded Controllers**
6. **Learns to simulate and configure different timer controls**

B. Sc (Information Technology)		Semester – II	
Course Name: Web Applications Development		Course Code: USIT203	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2	75
	Internal	--	25

Course Objectives:
- **Understand basic concepts of Internet and World Wide Web.**
- **Comprehend different HTML elements that can be used to develop static web pages.**
- **Become familiar with concept of stylesheets and various CSS effects.**
- **Peruse JavaScript as a tool to add dynamism to static HTML pages.**
- **Explore how server-side script works on the web.**
- **Learn how PHP can be connected to a database to store and retrieve data.**

Unit	Details	Lectures
I	**Internet and the World Wide Web**: What is Internet? Applications of Internet, E-mail, Telnet, FTP, E-commerce and E-business. Internet Service Providers, Domain Name Server, Internet Address, World Wide Web (WWW): World Wide Web and its Evolution, Uniform Resource Locator (URL), Browsers, Common Features of Browsers, Search Engine, Web Server, HTTP Protocol. **HTML5**: Introduction, Formatting Text by using Tags, Using Lists, Creating Hyperlinks and Bookmarks, Defining Metadata about an HTML Document, Redirecting to another URL. **CSS**: Implementing Styles using CSS – Stylesheets, Formatting Text and Links using CSS, CSS Selectors, Changing Background, Adding Border, Margin and Padding, Setting Dimensions, Using Inline Container to mark up a part of a text.	12
II	**HTML Page Layout**: Using Layout Elements, Semantic Elements, Creating, Positioning and Formatting Divisions, Floating Divisions next to each other, Responsive Web Design, Inline Frames. **HTML Media, Tables and Forms**: Embedding Images, Creating Client-side and Server-side Image Map, adding Favicon, Embedding audio and video on web page. Creating Simple Table, Table Dimension, Merging Table Cells, Formatting Tables: Applying Borders, Background and Foreground fills, Changing Cell Padding, Spacing and Alignment Collecting user input with HTML Forms, Additional Input Types in HTML5.	12
III	**JavaScript**: Introduction, Difference between Client-side and Server-side Scripting, JavaScript Variables and Constants, Data Types, Statements, Comments, Functions, Variable Scope, Hoisting, Strict Mode, JavaScript Objects, Dialog Boxes, void Keyword	12

	Operators: Arithmetic Operators, Assignment Operators, Comparison Operators, Logical Operators, Bitwise Operators **Statements**: Conditional Statements – if else, switch, Loops – while, do while, for, for in, for of, Loop Control – break, continue, labels **JavaScript Objects**: User-defined Objects, with Keyword, Native Objects – Array, String, Date, Math, Number, RegExp **DOM**: Introduction, DOM Properties and Methods. **Browser BOM**: Moving back and forward with History, Cookies **Events and Event Handlers**: HTML Events, DOM Events, DOM Event Listener, onAbort, onBlur, onChange, onClick, onDblClick, onError, onFocus, onKeyDown, onKeyPress, onKeyUp, onLoad, onMouseDown, onMouseMove, onMouseOut, onMouseOver, onMouseUp, onReset, onResize, onSelect, onSubmit, onUnload	
IV	**PHP**: Introduction, Server-side Scripting, PHP Syntax and Comments, Variables and Constants, Data Types, Control Structures, Looping, Loop Termination, Functions, PHP Form Handling, PHP Form Validation, Superglobals, PHP Arrays, PHP Strings, PHP RegEx, PHP Numbers, PHP Math, Basic PHP Errors	**12**
V	**Advanced PHP**: PHP Date and Time, PHP Include, PHP Cookies, PHP Sessions, Validating and Sanitizing Data with PHP Filters **PHP and MySQL** Why PHP and MySQL? Connect to MySQL, Creating Database and Tables, Inserting Single and Multiple Rows, Retrieving Last ID, MySQL Prepared, Selecting Data, Updating Data, Deleting Data, Limiting Data.	**12**

Course Outcomes:
- **Analyze working of Internet.**
- **Gain an insight into designing web pages.**
- **Use different ways of styling web pages using CSS.**
- **Implement basic and complex functionalities of JavaScript in a web page.**
- **Employ PHP Scripts to execute dynamic tasks in a web page.**
- **Perform various database tasks using PHP.**

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
	The Complete Reference HTML & CSS	Thomas A. Powell	McGrawHill	5th	-
	Step by Step HTML5	Faithe Wempen	Microsoft Press		2011
	Learning PHP, MySQL, JavaScript, CSS & HTML5	Robin Nixon	O'Reilly	3rd	2018
	Learning Web Design A Beginner's Guide to Html, CSS, JavaScript, And Web Graphics	Jennifer Niederst Robbins	O'Reilly	5th	2018

	The Complete Reference JavaScript	Thomas A. Powell & Fritz Schneider	McGrawHill	3rd	2012
	PHP & MySQL Novice to Ninja	Tom Butler	SPD	7th	2022

B. Sc (Information Technology)		**Semester – II**	
Course Name: Web Application Development Practical		**Course Code: USIT2P3**	
Periods per week (1 Period is 50 minutes)		**3**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Practical Examination**	**2½**	**50**
	Internal	**--**	**--**

Course Objectives:
- **Understand how to effectively implement HTML.**
- **Write CSS effectively to create well organized, styled web pages.**
- **Add versatility to a web page with client-side scripting.**
- **Deploy a local web server and run a simple web application.**
- **Read and process data in MySQL using PHP.**

List of Practical:	
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 that automatically redirects the user to another page.
2.	**Use of CSS**
a.	Design a web page demonstrating different stylesheet types.
b.	Design a web page demonstrating grouping selectors.
3.	**Layout and Media**
a.	Design a web page demonstrating different semantics.
b.	Design a web page embedding image, audio and video.
c.	Design a web page with Imagemaps.
4.	**Tables and Forms**
a.	Design a web page with different tables.
b.	Design a web page with a form that uses all types of controls.

5.	**JavaScript**
a.	Using JavaScript, design a web page to accept a number from the user and print its Factorial.
b.	Using JavaScript, a web page that prints Fibonacci series/any given series.
c.	Write a JavaScript program to display all the prime numbers between 1 and 100.
d.	Write a JavaScript program to accept a number from the user and display the sum of its digits.
6.	**JavaScript Objects**
a.	Using JavaScript, design a web page demonstrating different native objects of JavaScript.
b.	Write a program in JavaScript to accept a sentence from the user and display the number of words in it. (Do not use split () function).
7.	**JavaScript Events**
a.	Write a JavaScript program to design simple calculator.
b.	Design a form and validate all the controls placed on the form using JavaScript.
8.	**Basic PHP**
a.	Write a PHP code to find the greater of 2 numbers. Accept the no. from the user.
b.	Write a PHP Program to accept a number from the user and print it factorial.
c.	Write a PHP program to accept a number from the user and print whether it is prime or not.
d.	Write a PHP program to display the following Binary Pyramid: 1 0 1 1 0 1 0 1 0 1 1 0 1 0 1
e.	Write a PHP program to demonstrate different string functions.
f.	Write a PHP program to demonstrate different array functions.
9.	**Advanced PHP**
a.	Write a PHP program to demonstrate use of sessions and cookies.
b.	Write a PHP program to demonstrate use of filters.
10.	**PHP and MySQL**
a.	Write a PHP program 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 format.

c.	Write a PHP program to
	    Update rows in a table
	    Delete rows from a table

d.	Design a PHP page for authenticating a user

Course Outcomes:

Learners will be able to,

1. **Design static web pages using Hyper Text Markup Language (HTML).**
2. **Enhance the look of web pages by implementing CSS.**
3. **Collect information from the user with HTML Forms.**
4. **Design interactive webpages using client-side script (JavaScript).**
5. **Implement Document Object Model and events in web pages using JavaScript.**
6. **Write and deploy basic PHP code to simplify web development.**
7. **Store and retrieve data from a server using PHP**.

B. Sc (Information Technology)	Semester – I		
Course Name: Numerical Methods	Course Code: USIT204		
Periods per week (1 Period is 50 minutes)	5		
Credits	2		
		Hours	Marks
Evaluation System	Theory Examination	2	75
	Internal	--	25

Course Objectives:
Course will enhance the problem solving skills of students using extremely powerful numerical methods.

Unit	Details	Lectures
I	**Mathematical Modeling and Engineering Problem Solving:** A Simple Mathematical Model, Conservation Laws and Engineering Problems **Approximations and Round-Off Errors:** Significant Figures, Accuracy and Precision, Error Definitions, Round-Off Errors **Truncation Errors and the Taylor Series:** The Taylor Series, Error Propagation, Total Numerical Errors, Formulation Errors and Data Uncertainty	12
II	**Solutions of Algebraic and Transcendental Equations:** The Bisection Method, The Newton-Raphson Method, The Regula-falsi method, The Secant Method. **Interpolation:** Forward Difference, Backward Difference, Newton's Forward Difference Interpolation, Newton's Backward Difference Interpolation, Lagrange's Interpolation	12
III	**Solution of simultaneous algebraic equations (linear) using iterative methods:** Gauss-Jordan Method, Gauss-Seidel Method. **Numerical differentiation and Integration:** Numberical differentiation, Numerical integration using Trapezoidal Rule, Simpson's $1/3_{rd}$ and $3/8_{th}$ rules.	12
IV	**Numerical solution of 1st and 2nd order differential equations:** Taylor series, Euler's Method, Modified Euler's Method, Runge-Kutta Method for 1_{st} and 2_{nd} Order Differential Equations. **Least-Squares Regression:** Linear Regression, Polynomial Regression, Multiple Linear Regression, General Linear Least Squares, Nonlinear Regression	12
V	**Linear Programming:** Linear optimization problem, Formulation and Graphical solution, Basic solution and Feasible solution. **Numerical Solutions of Partial Differential Equations:** Classification of Partial Differential Equations of the second order, Difference equation corresponding to Laplace Equation, Liebmann's Iteration Process, Bender-Schmidt's Difference Equation corresponding to the parabolic equation, Crank Nicolson's difference equations corresponding to the parabolic equation, Difference equation corresponding to the Hyperbolic equation..	12

Books and References:					
Sr. No.	Title	Author/s	Publisher	Edition	Year

1.	Introductory Methods of Numerical Methods	S. S. Sastry	PHI	5th	2012
2.	Numerical Methods for Engineers	Steven C. Chapra, Raymond P. Canale	Tata Mc Graw Hill	6th	2010
3.	Numerical Analysis	Richard L. Burden, J. Douglas Faires	Cengage Learning	9th	2011
4.	Numerical Methods	T Veerarajan T Ramachandran	Tata Mc Graw Hill	7th	2011

Course Outcomes:

Learners will be able to,

1. **Understand numerical techniques to find the roots of non-linear equations and solution of system of linear equations.**
2. **Understand the difference operators and the use of interpolation.**
3. **Understand numerical differentiation and integration and numerical solutions of ordinary and partial differential equations.**

B. Sc (Information Technology)	**Semester – I**		
Course Name: Numerical Methods Practical	**Course Code: USIT2P4**		
Periods per week (1 Period is 50 minutes)	**3**		
Credits	**2**		
		Hours	**Marks**
Evaluation System	**Practical Examination**	2½	50
	Internal	--	--

Course Objectives:

Course will provide different tools to find solutions to various numerical techniques

List of Practical: To be implemented using SCILAB	
1.	**Iterative Calculation**
a.	Program for iterative calculation
b.	Program to calculate the roots of a quadratic equation using the formula
c.	Program to evaluate e^x using infinite series
2.	**Solution of algebraic and transcendental equations:**

a.	Program to solve algebraic and transcendental equation 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.
3.	**Interpolation**
a.	Program for Newton's forward interpolation.
b.	Program for Newton's backward interpolation.
c.	Program for Lagrange's interpolation.
4.	**Solving linear system of equations by iterative methods**
a.	Program for solving linear system of equations using Gauss Jordan method.
b.	Program for solving linear system of equations using Gauss Seidel method.
5.	**Numerical Differentiation**
a.	Program to obtain derivatives numerically.
6.	**Numerical Integration**
a.	Program for numerical integration using Trapezoidal rule.
b.	Program for numerical integration using Simpson's $1/3_{rd}$ rule.
c.	Program for numerical integration using Simpson's $3/8_{th}$ rule.
7.	**Solution of differential equations**
a.	Program to solve differential equation using Euler's method.
b.	Program to solve differential equation using modified Euler's method.
c.	Program to solve differential equation using Runge-kutta 2_{nd} order and 4_{th} order methods.
8.	**Regression** 1
a.	Program for Linear regression.
b.	Program for Polynomial Regression.
9.	**Regression** 2

a.	Program for multiple linear regression.
b.	Program for non-linear regression.
10.	Numerical solution of partial differential equations
a.	Program to find solution of Laplace's equation.

Course Outcomes:

Learners will be able to,

1. **Find fast and accurate solution to simple and complex numerical problems using these programs.**

B. Sc (Information Technology)		Semester – II	
Course Name: Green IT		**Course Code: USIT205**	
Periods per week (1 Period is 50 minutes)		**5**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	2	75
	Internal	--	25

Course Objectives:
- **To understand the concept of Green Technology.**
- **To learn Green IT regulating Green IT and different standards.**
- **To understand the concept of minimizing power utilization in technology.**
- **To know about Green PCs, Green notebooks and servers and Green data centers.**
- **To know how the way of work is changing and understand implementation of Paperless work.**
- **To know the concept of Recycling.**
- **To understand Metrics for Green IT.**

Unit	Details	Lectures
I	**Overview to Green IT:** Problems: Toxins, Power Consumption, Equipment Disposal, Company's Carbon Footprint: Measuring, Details, reasons to bother, Plan for the Future, Cost Savings: Hardware, Power. **Regulating Green IT: Laws, Standards and Protocols** Introduction, The Regulatory Environment and IT Manufacturers RoHS, REACh, WEEE, Legislating for GHG Emissions and Energy Use of IT Equipment.Nonregulatory Government Initiatives, Industry Associations and Standards Bodies, Green Building Standards, Green Data Centres , Social Movements and Greenpeace.	**12**
II	**Minimizing Power Usage:** Power Problems, Monitoring Power Usage, Servers, Low-CostOptions, Reducing Power Use, Data De-Duplication, Virtualization, Management, Bigger Drives, Involving the Utility Company, LowPower Computers, PCs, Linux, Components, Servers, ComputerSettings, Storage, Monitors, Power Supplies, Wireless Devices, Software. **Cooling:** Cooling Costs, Power Cost, Causes of Cost, Calculating CoolingNeeds, Reducing Cooling Costs, Economizers, On-Demand Cooling, HP's Solution, Optimizing Airflow, Hot Aisle/Cold Aisle, Raised, Floors, Cable Management, Vapour Seal, Prevent Recirculation of Equipment Exhaust, Supply Air Directly to Heat Sources, Fans, Humidity, Adding Cooling, Fluid Considerations, System Design, Datacentre Design, Centralized Control, Design for Your Needs, Put Everything Together.	**12**
III	**Greening IT:** Green PCs, Notebooks and Servers, Green Data Centres , Green Cloud Computing, Green Data Storage , Green Software, Green Networking and Communications. **Changing the Way of Work:** Old Behaviours, starting at the Top, Process Reengineering with Green in Mind, Analysing the Global Impact of Local Actions, Steps: Water, Recycling, Energy, Pollutants, Teleworkers and Outsourcing, Telecommuting, Outsourcing, how to Outsource. **Going Paperless:** Paper Problems, The Environment, Costs: Paper and Office, Practicality, Storage, Destruction, Going Paperless, Organizational Realities, Changing Over, Paperless Billing, Handheld Computers vs. the Clipboard, Unified Communications, Intranets, What to Include, Building an Intranet, Microsoft Office SharePoint Server 2007, Electronic Data Interchange (EDI), Nuts and Bolts, Value Added Networks, Advantages, Obstacles.	**12**

IV	**Recycling:** Means of Disposal, Recycling, Refurbishing, Make the Decision, Life Cycle, from beginning to end, Life, Cost, Green Design, Recycling Companies, Finding the Best One, Checklist, Certifications, Hard Drive Recycling, Consequences, cleaning a Hard Drive, Pros and cons of each method, CDs and DVDs, good and bad about CD and DVDs disposal, Change the mind-set, David vs. America Online. **Hardware Considerations:** Certification Programs, EPEAT, RoHS, Energy Star, Computers, Monitors, Printers, Scanners, All-in-Ones, Thin Clients, Servers, Blade Servers, Consolidation, Products, Hardware Considerations, Planned Obsolescence, Packaging, Toxins, Other Factors, Remote Desktop, Using Remote Desktop, Establishing a Connection.	**12**
V	**Greening Your Information Systems:** Initial Improvement Calculations, Selecting Metrics, Tracking Progress, Change Business Processes, Customer Interaction, Paper Reduction, Green Supply Chain, Improve Technology Infrastructure, Reduce PCs and Servers, Shared Services, Hardware Costs, Cooling. **Staying Green:** Organizational Check-ups, Chief Green Officer, Evolution, Sell the CEO, SMART Goals, Equipment Check-ups, Gather Data, Tracking the data, Baseline Data, Benchmarking, Analyse Data, Conduct Audits, Certifications, Benefits, Realities, Helpful Organizations.	**12**

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Green IT	Toby Velte, Anthony Velte, Robert Elsenpeter	McGraw Hill		2008
2.	Harnessing Green IT: Principles and Practices	San Murugesan, G. R. Ganadharan,	Wiley & IEEE.		
3.	Green Data Center: Steps for the Journey	Alvin Galea, Michael Schaefer, Mike Ebbers	Shroff Publishers and Distributers		2011
4.	Green IT	Deepak Shikarpur	Vishwkarma Publications,		2014
5.	Green Computing Tools and Techniques for Saving Energy, Money and Resources	Bud E. Smith	CRC Press		2014
	Green Computing and Green IT Best Practice	Jason Harris	Emereo		

Course Outcomes:

Learners will be able to,

- **Understand the concept of Green IT and problems related to it.**
- **Know different standards for Green IT.**
- **Understand the how power usage can be minimized in Technology.**
- **Learn about how the way of work is changing.**
- **Understand the concept of recycling.**
- **Know how information system can stay Green Information system.**

B. Sc (Information Technology)		**Semester – II**	
Course Name: Practical's in PL/SQL		Course Code: USIT2P5	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	**Marks**
Evaluation System	Practical Examination	2½	50
	Internal	--	--

Course Objectives:
- **To understand the basics of PL/SQL.**
- **To understand control and conditional statement in PL/SQL.**
- **To understand working of sequences and cursor in PL/SQL.**
- **To understand concept of stored procedure and functions.**
- **To understand triggers and packages in PL/SQL.**
- **To understand the concept of Exception handling.**

List of Practical:	
1.	**PL/SQL Basics**
a.	Use of variables.
b.	Write executable statement.
c.	Interacting with Oracle Server.
d.	Create anonymous PL/SQL block
2.	**Control Structure in PL/SQL .**
a.	Using while loop
b.	Do loop
c.	For loop
d.	Use of GOTO statement
3.	**Create conditional statement using PL/SQL**
a.	Using if statement
b.	Using if else statement
c.	Using elsif ladder
d.	Using case expression.
4.	**Creation of Sequence in PL/SQL**
5.	**Create cursor in PL/SQL**
a.	Implicit cursor
b.	Explicit

c.	Parameterized cursor
d.	Cursor for loop
6.	**Creation of Procedures in PL/SQL**
7.	**Functions in PL/SQL**
a.	Compute and returns the maximum value
b.	Compute factorial of given number.
8.	**Creation of Trigger**
a.	Create Row level trigger
b.	Create Statement level trigger
c.	Create instead of trigger
9.	**Handling exceptions**
a.	Creation of user defined exception
b.	Creation of system defined exception.
10.	**Creation of Package in PL/SQL**

Course Outcomes:

Learner will be able to:

- **Understand the basics of PL/SQL.**
- **Use of the control and conditional statement in PL/SQL.**
- **Apply sequences and cursor in PL/SQL.**
- **Know the concept of stored procedure and functions**
- **Create the triggers and packages in PL/SQL.**
- **Implement the concept of Exception handling.**

UNIVERSITY OF MUMBAI

Syllabus for S.Y.B.Sc.

Programme: B.Sc.

Course: Information Technology

with effect from the academic year

2017 – 2018

Semester – 3			
Course Code	**Course Type**	**Course Title**	**Credits**
USIT301	Skill Enhancement Course	Python Programming	2
USIT302	Core Subject	Data Structures	2
USIT303	Core Subject	Computer Networks	2
USIT304	Core Subject	Database Management Systems	2
USIT305	Core Subject	Applied Mathematics	2
USIT3P1	Skill Enhancement Course Practical	Python Programming Practical	2
USIT3P2	Core Subject Practical	Data Structures Practical	2
USIT3P3	Core Subject Practical	Computer Networks Practical	2
USIT3P4	Core Subject Practical	Database Management Systems Practical	2
USIT3P5	Core Subject Practical	Mobile Programming Practical	2
		Total Credits	**20**

Semester – 4			
Course Code	**Course Type**	**Course Title**	**Credits**
USIT401	Skill Enhancement Course	Core Java	2
USIT402	Core Subject	Introduction to Embedded Systems	2
USIT403	Core Subject	Computer Oriented Statistical Techniques	2
USIT404	Core Subject	Software Engineering	2
USIT405	Core Subject	Computer Graphics and Animation	2
USIT4P1	Skill Enhancement Course Practical	Core Java Practical	2
USIT4P2	Core Subject Practical	Introduction to Embedded Systems Practical	2
USIT4P3	Core Subject Practical	Computer Oriented Statistical Techniques Practical	2
USIT4P4	Core Subject Practical	Software Engineering Practical	2
USIT4P5	Core Subject Practical	Computer Graphics and Animation Practical	2
		Total Credits	**20**

SEMESTER III

B. Sc. (Information Technology)		Semester – III	
Course Name: Python Programming		**Course Code: USIT301**	
Periods per week (1 Period is 50 minutes)		**5**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	**2½**	**75**
	Internal	**--**	**25**

Unit	Details	Lectures
I	**Introduction:** The Python Programming Language, History, features, Installing Python, Running Python program, Debugging : Syntax Errors, Runtime Errors, Semantic Errors, Experimental Debugging, Formal and Natural Languages, The Difference Between Brackets, Braces, and Parentheses, **Variables and Expressions** Values and Types, Variables, Variable Names and Keywords, Type conversion, Operators and Operands, Expressions, Interactive Mode and Script Mode, Order of Operations. **Conditional Statements:** if, if-else, nested if –else **Looping**: for, while, nested loops **Control statements:** Terminating loops, skipping specific conditions	**12**
II	**Functions:** Function Calls, Type Conversion Functions, Math Functions, Composition, Adding New Functions, Definitions and Uses, Flow of Execution, Parameters and Arguments, Variables and Parameters Are Local, Stack Diagrams, Fruitful Functions and Void Functions, Why Functions? Importing with from, Return Values, Incremental Development, Composition, Boolean Functions, More Recursion, Leap of Faith, Checking Types **Strings:** A String Is a Sequence, Traversal with a for Loop, String Slices, Strings Are Immutable, Searching, Looping and Counting, String Methods, The in Operator, String Comparison, String Operations.	**12**
III	**Lists:** Values and Accessing Elements, Lists are mutable, traversing a List, Deleting elements from List, Built-in List Operators, Concatenation, Repetition, In Operator, Built-in List functions and methods **Tuples and Dictionaries**: Tuples, Accessing values in Tuples, Tuple Assignment, Tuples as return values, Variable-length argument tuples, Basic tuples operations, Concatenation, Repetition, in Operator, Iteration, Built-in Tuple Functions Creating a Dictionary, Accessing Values in a dictionary, Updating Dictionary, Deleting Elements from Dictionary, Properties of Dictionary keys, Operations in Dictionary, Built-In Dictionary Functions, Built-in Dictionary Methods **Files:** Text Files, The File Object Attributes, Directories	**12**

		Exceptions: Built-in Exceptions, Handling Exceptions, Exception with Arguments, User-defined Exceptions	
IV		**Regular Expressions –** Concept of regular expression, various types of regular expressions, using match function. **Classes and Objects:** Overview of OOP (Object Oriented Programming), Class Definition, Creating Objects, Instances as Arguments, Instances as return values, Built-in Class Attributes, Inheritance, Method Overriding, Data Encapsulation, Data Hiding **Multithreaded Programming:** Thread Module, creating a thread, synchronizing threads, multithreaded priority queue **Modules:** Importing module, Creating and exploring modules, Math module, Random module, Time module	12
V		**Creating the GUI Form and Adding Widgets:** **Widgets:** Button, Canvas, Checkbutton, Entry, Frame, Label, Listbox, Menubutton, Menu, Message, Radiobutton, Scale, Scrollbar, text, Toplevel, Spinbox, PanedWindow, LabelFrame, tkMessagebox. Handling Standard attributes and Properties of Widgets. **Layout Management:** Designing GUI applications with proper Layout Management features. **Look and Feel Customization**: Enhancing Look and Feel of GUI using different appearances of widgets. **Storing Data in Our MySQL Database via Our GUI :** Connecting to a MySQL database from Python, Configuring the MySQL connection, Designing the Python GUI database, Using the INSERT command, Using the UPDATE command, Using the DELETE command, Storing and retrieving data from MySQL database.	12

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Think Python	Allen Downey	O'Reilly	1st	2012
2.	An Introduction to Computer Science using Python 3	Jason Montojo, Jennifer Campbell, Paul Gries	SPD	1st	2014
3.	Python GUI Programming Cookbook	Burkhard A. Meier	Packt		2015
4.	Introduction to Problem Solving with Python	E. Balagurusamy	TMH	1st	2016
5.	Murach's Python programming	Joel Murach, Michael Urban	SPD	1st	2017
6.	Object-oriented Programming in Python	Michael H. Goldwasser, David Letscher	Pearson Prentice Hall	1st	2008
7.	Exploring Python	Budd	TMH	1st	2016

B. Sc. (Information Technology)		Semester – III	
Course Name: Data Structures		Course Code: USIT302	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2½	75
	Internal	--	25

Unit	Details	Lectures
I	**Introduction:** Data and Information, Data Structure, Classification of Data Structures, Primitive Data Types, Abstract Data Types, Data structure vs. File Organization, Operations on Data Structure, Algorithm, Importance of Algorithm Analysis, Complexity of an Algorithm, Asymptotic Analysis and Notations, Big O Notation, Big Omega Notation, Big Theta Notation, Rate of Growth and Big O Notation. **Array**: Introduction, One Dimensional Array, Memory Representation of One Dimensional Array, Traversing, Insertion, Deletion, Searching, Sorting, Merging of Arrays, Multidimensional Arrays, Memory Representation of Two Dimensional Arrays, General Multi-Dimensional Arrays, Sparse Arrays, Sparse Matrix, Memory Representation of Special kind of Matrices, Advantages and Limitations of Arrays.	12
II	**Linked List:** Linked List, One-way Linked List, Traversal of Linked List, Searching, Memory Allocation and De-allocation, Insertion in Linked List, Deletion from Linked List, Copying a List into Other List, Merging Two Linked Lists, Splitting a List into Two Lists, Reversing One way linked List, Circular Linked List, Applications of Circular Linked List, Two way Linked List, Traversing a Two way Linked List, Searching in a Two way linked List, Insertion of an element in Two way Linked List, Deleting a node from Two way Linked List, Header Linked List, Applications of the Linked list, Representation of Polynomials, Storage of Sparse Arrays, Implementing other Data Structures.	12
III	**Stack**: Introduction, Operations on the Stack Memory Representation of Stack, Array Representation of Stack, Applications of Stack, Evaluation of Arithmetic Expression, Matching Parenthesis, infix and postfix operations, Recursion**.** **Queue:** Introduction, Queue, Operations on the Queue, Memory Representation of Queue, Array representation of queue, Linked List Representation of Queue, Circular Queue, Some special kinds of queues, Deque, Priority Queue, Application of Priority Queue, Applications of Queues.	12

IV	**Sorting and Searching Techniques** Bubble, Selection, Insertion, Merge Sort. Searching: Sequential, Binary, Indexed Sequential Searches, Binary Search. **Tree:** Tree, Binary Tree, Properties of Binary Tree, Memory Representation of Binary Tree, Operations Performed on Binary Tree, Reconstruction of Binary Tree from its Traversals, Huffman Algorithm, Binary Search Tree, Operations on Binary Search Tree, Heap, Memory Representation of Heap, Operation on Heap, Heap Sort. **Advanced Tree Structures:** Red Black Tree, Operations Performed on Red Black Tree, AVL Tree, Operations performed on AVL Tree, 2-3 Tree, B-Tree.	12
V	**Hashing Techniques** Hash function, Address calculation techniques, Common hashing functions Collision resolution, Linear probing, Quadratic, Double hashing, Bucket hashing, Deletion and rehashing **Graph**: Introduction, Graph, Graph Terminology, Memory Representation of Graph, Adjacency Matrix Representation of Graph, Adjacency List or Linked Representation of Graph, Operations Performed on Graph, Graph Traversal, Applications of the Graph, Reachability, Shortest Path Problems, Spanning Trees.	12

Books and References:					
Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	A Simplified Approach to Data Structures	Lalit Goyal, Vishal Goyal, Pawan Kumar	SPD	1st	2014
2.	An Introduction to Data Structure with Applications	Jean – Paul Tremblay and Paul Sorenson	Tata MacGraw Hill	2nd	2007
3.	Data Structure and Algorithm	Maria Rukadikar	SPD	1st	2017
4.	Schaum's Outlines Data structure	Seymour Lipschutz	Tata McGraw Hill	2nd	2005
5.	Data structure – A Pseudocode Approach with C	AM Tanenbaum, Y Langsam and MJ Augustein	Prentice Hall India	2nd	2006
6.	Data structure and Algorithm Analysis in C	Weiss, Mark Allen	Addison Wesley	1st	2006

B. Sc. (Information Technology)		Semester – III	
Course Name: Computer Networks		**Course Code: USIT303**	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	**Marks**
Evaluation System	**Theory Examination**	2½	75
	Internal	--	25

Unit	Details	Lectures
I	**Introduction:** Data communications, networks, network types, Internet history, standards and administration. **Network Models:** Protocol layering, TCP/IP protocol suite, The OSI model. **Introduction to Physical layer:** Data and signals, periodic analog signals, digital signals, transmission impairment, data rate limits, performance. **Digital and Analog transmission:** Digital-to-digital conversion, analog-to-digital conversion, transmission modes, digital-to-analog conversion, analog-to-analog conversion.	**12**
II	**Bandwidth Utilization: Multiplexing and Spectrum Spreading:** Multiplexing, Spread Spectrum **Transmission media:** Guided Media, Unguided Media **Switching:** Introduction, circuit switched networks, packet switching, structure of a switch. **Introduction to the Data Link Layer:** Link layer addressing, Data Link Layer Design Issues, Error detection and correction, block coding, cyclic codes, checksum, forward error correction, error correcting codes, error detecting codes.	**12**
III	**Data Link Control:** DLC services, data link layer protocols, HDLC, Point-to-point protocol. **Media Access Control**: Random access, controlled access, channelization, Wired LANs – Ethernet Protocol, standard ethernet, fast ethernet, gigabit ethernet, 10 gigabit ethernet, **Wireless LANs:** Introduction, IEEE 802.11 project, Bluetooth, WiMAX, Cellular telephony, Satellite networks. **Connecting devices and Virtual LANs.**	**12**
IV	**Introduction to the Network Layer**: Network layer services, packet switching, network layer performance, IPv4 addressing, forwarding of IP packets, Internet Protocol, ICMPv4, Mobile IP **Unicast Routing:** Introduction, routing algorithms, unicast routing protocols. **Next generation IP:** IPv6 addressing, IPv6 protocol, ICMPv6 protocol, transition from IPv4 to IPv6.	**12**
V	**Introduction to the Transport Layer**: Introduction, Transport layer protocols (Simple protocol, Stop-and-wait protocol, Go-Back-n	**12**

protocol, Selective repeat protocol, Bidirectional protocols), Transport layer services, User datagram protocol, Transmission control protocol, **Standard Client0Server Protocols:** World wide-web and HTTP, FTP, Electronic mail, Telnet, Secured Shell, Domain name system.	

Books and References:					
Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Data Communication and Networking	Behrouz A. Forouzan	Tata McGraw Hill	Fifth Edition	2013
2.	TCP/IP Protocol Suite	Behrouz A. Forouzan	Tata McGraw Hill	Fourth Edition	2010
3.	Computer Networks	Andrew Tanenbaum	Pearson	Fifth	2013

B. Sc. (Information Technology)		Semester – III	
Course Name: Database Management Systems		**Course Code: USIT304**	
Periods per week (1 Period is 50 minutes)		**5**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	2½	75
	Internal	--	25

Unit	Details	Lectures
I	**Introduction to Databases and Transactions** 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. **Database Design, ER Diagram and Unified Modeling Language** Database design and ER Model: overview, ER Model, Constraints, ER Diagrams, ERD Issues, weak entity sets, Codd's rules, Relational Schemas, Introduction to UML	**12**
II	**Relational database model:** Logical view of data, keys, integrity rules, Relational Database design: features of good relational database design, atomic domain and Normalization (1NF, 2NF, 3NF, BCNF). **Relational Algebra and Calculus** Relational algebra: introduction, Selection and projection, set operations, renaming, Joins, Division, syntax, semantics. Operators, grouping and ungrouping, relational comparison. **Calculus:** Tuple relational calculus, Domain relational Calculus, calculus vs algebra, computational capabilities	**12**
III	**Constraints, Views and SQL** 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, Null Values, nested sub queries, Joined relations. Triggers.	**12**
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.	**12**
V	**PL-SQL**: Beginning with PL / SQL, Identifiers and Keywords, Operators, Expressions, Sequences, Control Structures, Cursors and Transaction, Collections and composite data types, Procedures and Functions, Exceptions Handling, Packages, With Clause and Hierarchical Retrieval, Triggers.	**12**

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Database System and Concepts	A Silberschatz, H Korth, S Sudarshan	McGraw-Hill	Fifth Edition	
2.	Database Systems	Rob Coronel	Cengage Learning	Twelfth Edition	
3.	Programming with PL/SQL for Beginners	H. Dand, R. Patil and T. Sambare	X –Team	First	2011
4.	Introduction to Database System	C.J.Date	Pearson	First	2003

B. Sc. (Information Technology)		Semester – III	
Course Name: Applied Mathematics		**Course Code: USIT305**	
Periods per week (1 Period is 50 minutes)		**5**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	**2½**	**75**
	Internal	**--**	**25**

Unit	Details	Lectures
I	**Matrices:** Inverse of a matrix, Properties of matrices, Elementary Transformation, Rank of Matrix, Echelon or Normal Matrix, Inverse of matrix, Linear equations, Linear dependence and linear independence of vectors, Linear transformation, Characteristics roots and characteristics vectors, Properties of characteristic vectors, Caley-Hamilton Theorem, Similarity of matrices, Reduction of matrix to a diagonal matrix which has elements as characteristics values. **Complex Numbers:** Complex number, Equality of complex numbers, Graphical representation of complex number(Argand's Diagram), Polar form of complex numbers, Polar form of x+iy for different signs of x,y, Exponential form of complex numbers, Mathematical operation with complex numbers and their representation on Argand's Diagram, Circular functions of complex angles, Definition of hyperbolic function, Relations between circular and hyperbolic functions, Inverse hyperbolic functions, Differentiation and Integration, Graphs of the hyperbolic functions, Logarithms of complex quality, j(=i)as an operator(Electrical circuits)	**12**
II	**Equation of the first order and of the first degree:** Separation of variables, Equations homogeneous in x and y, Non-homogeneous linear equations, Exact differential Equation, Integrating Factor, Linear Equation and equation reducible to this form, Method of substitution. **Differential equation of the first order of a degree higher than the first:** Introduction, Solvable for p (or the method of factors), Solve for y, Solve for x, Clairaut's form of the equation, Methods of Substitution, Method of Substitution. **Linear Differential Equations with Constant Coefficients:** Introduction, The Differential Operator, Linear Differential Equation $f(D) y = 0$, Different cases depending on the nature of the root of the equation $f(D) = 0$, Linear differential equation $f(D) y = X$, The complimentary Function, The inverse operator $1/f(D)$ and the symbolic expiration for the particular integral $1/f(D) X$; the general methods, Particular integral : Short methods, Particular integral : Other methods, Differential equations reducible to the linear differential equations with constant coefficients.	**12**
III	**The Laplace Transform:** Introduction, Definition of the Laplace Transform, Table of Elementary Laplace Transforms, Theorems on Important Properties of Laplace Transformation, First Shifting	**12**

	Theorem, Second Shifting Theorem, The Convolution Theorem, Laplace Transform of an Integral, Laplace Transform of Derivatives, **Inverse Laplace Transform:** Shifting Theorem, Partial fraction Methods, Use of Convolution Theorem, Solution of Ordinary Linear Differential Equations with Constant Coefficients, Solution of Simultaneous Ordinary Differential Equations, Laplace Transformation of Special Function, Periodic Functions, Heaviside Unit Step Function, Dirac-delta Function(Unit Impulse Function),	
IV	**Multiple Integrals:** Double Integral, Change of the order of the integration, Double integral in polar co-ordinates, Triple integrals. **Applications of integration:** Areas, Volumes of solids.	**12**
V	**Beta and Gamma Functions –** Definitions**,** Properties and Problems. Duplication formula. **Differentiation Under the Integral Sign** **Error Functions**	**12**

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	A text book of Applied Mathematics Vol I	P. N. Wartikar and J. N. Wartikar	Pune Vidyathi Graha		
2.	Applied Mathematics II	P. N. Wartikar and J. N. Wartikar	Pune Vidyathi Graha		
3.	Higher Engineering Mathematics	Dr. B. S. Grewal	Khanna Publications		

B. Sc. (Information Technology)		Semester – III	
Course Name: Python Programming Practical		**Course Code: USIT3P1**	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	**Marks**
Evaluation System	**Practical Examination**	**2½**	**50**
	Internal	**--**	**--**

List of Practical	
1.	**Write the program for the following:**
a.	Create a program that asks the user to enter their name and their age. Print out a message addressed to them that tells them the year that they will turn 100 years old.
b.	Enter the number from the user and depending on whether the number is even or odd, print out an appropriate message to the user.
c.	Write a program to generate the Fibonacci series.
d.	Write a function that reverses the user defined value.
e.	Write a function to check the input value is Armstrong and also write the function for Palindrome.
f.	Write a recursive function to print the factorial for a given number.
2.	**Write the program for the following:**
a.	Write a function that takes a character (i.e. a string of length 1) and returns True if it is a vowel, False otherwise.
b.	Define a function that computes the *length* of a given list or string.

c.	Define a *procedure* `histogram()` that takes a list of integers and prints a histogram to the screen. For example, `histogram([4, 9, 7])` should print the following:
	
	`****`
	`*********`
	`*******`

3.	**Write the program for the following:**
a.	A *pangram* is a sentence that contains all the letters of the English alphabet at least once, for example: *The quick brown fox jumps over the lazy dog*. Your task here is to write a function to check a sentence to see if it is a pangram or not.
b.	Take a list, say for example this one: `  a = [1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89]` and write a program that prints out all the elements of the list that are less than 5.
4.	**Write the program for the following:**

a.	Write a program that takes two lists and returns True if they have at least one common member.
b.	Write a Python program to print a specified list after removing the 0th, 2nd, 4th and 5th elements.
c.	Write a Python program to clone or copy a list
5.	**Write the program for the following:**
a.	Write a Python script to sort (ascending and descending) a dictionary by value.
b.	Write a Python script to concatenate following dictionaries to create a new one. Sample Dictionary : dic1={1:10, 2:20} dic2={3:30, 4:40} dic3={5:50,6:60} Expected Result : {1: 10, 2: 20, 3: 30, 4: 40, 5: 50, 6: 60}
c.	Write a Python program to sum all the items in a dictionary.
6.	**Write the program for the following:**
a.	Write a Python program to read an entire text file.
b.	Write a Python program to append text to a file and display the text.
c.	Write a Python program to read last n lines of a file.
7.	**Write the program for the following:**
a.	Design a class that store the information of student and display the same
b.	Implement the concept of inheritance using python
c.	Create a class called `Numbers`, which has a single class attribute called `MULTIPLIER`, and a constructor which takes the parameters `x` and `y` (these should all be numbers).
	i. Write a method called `add` which returns the sum of the attributes `x` and `y`.
	ii. Write a class method called multiply, which takes a single number parameter a and returns the product of a and MULTIPLIER.
	iii. Write a static method called subtract, which takes two number parameters, b and c, and returns b - c.
	iv. Write a method called value which returns a tuple containing the values of x and y. Make this method into a property, and write a setter and a deleter for manipulating the values of `x` and `y`.
8.	**Write the program for the following:**
a.	Open a new file in IDLE ("New Window" in the "File" menu) and save it as geometry.py in the directory where you keep the files you create for this course. Then copy the functions you wrote for calculating volumes and areas in the "Control Flow and Functions" exercise into this file and save it. Now open a new file and save it in the same directory. You should now be able to import your own module like this: `import geometry`

		Try and add `print dir(geometry)` to the file and run it. Now write a function pointyShapeVolume(x, y, squareBase) that calculates the volume of a square pyramid if squareBase is True and of a right circular cone if squareBase is False. x is the length of an edge on a square if squareBase is True and the radius of a circle when squareBase is False. y is the height of the object. First use squareBase to distinguish the cases. Use the circleArea and squareArea from the geometry module to calculate the base areas.
	b.	Write a program to implement exception handling.
9.		**Write the program for the following:**
	a.	Try to configure the widget with various options like: bg="red", family="times", size=18
	b.	Try to change the widget type and configuration options to experiment with other widget types like Message, Button, Entry, Checkbutton, Radiobutton, Scale etc.
10.		**Design the database applications for the following:**
	a.	Design a simple database application that stores the records and retrieve the same.
	b.	Design a database application to search the specified record from the database.
	c.	Design a database application to that allows the user to add, delete and modify the records.

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Think Python	Allen Downey	O'Reilly	1st	2012
2.	An Introduction to Computer Science using Python 3	Jason Montojo, Jennifer Campbell, Paul Gries	SPD	1st	2014

B. Sc. (Information Technology)		Semester – III	
Course Name: Data Structures Practical		Course Code: USIT3P2	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	**Marks**
Evaluation System	Practical Examination	2½	50
	Internal	--	--

List of Practical	
1.	**Implement the following:**
a.	Write a program to store the elements in 1-D array and perform the operations like searching, sorting and reversing the elements. [Menu Driven]
b.	Read the two arrays from the user and merge them and display the elements in sorted order.[Menu Driven]
c.	Write a program to perform the Matrix addition, Multiplication and Transpose Operation. [Menu Driven]
2.	**Implement the following for Linked List:**
a.	Write a program to create a single linked list and display the node elements in reverse order.
b.	Write a program to search the elements in the linked list and display the same
c.	Write a program to create double linked list and sort the elements in the linked list.
3.	**Implement the following for Stack:**
a.	Write a program to implement the concept of Stack with Push, Pop, Display and Exit operations.
b.	Write a program to convert an infix expression to postfix and prefix conversion.
c.	Write a program to implement Tower of Hanoi problem.
4.	**Implement the following for Queue:**
a.	Write a program to implement the concept of Queue with Insert, Delete, Display and Exit operations.
b.	Write a program to implement the concept of Circular Queue
c.	Write a program to implement the concept of Deque.
5.	**Implement the following sorting techniques:**
a.	Write a program to implement bubble sort.
b.	Write a program to implement selection sort.
c.	Write a program to implement insertion sort.
6.	**Implement the following data structure techniques:**
a.	Write a program to implement merge sort.
b.	Write a program to search the element using sequential search.
c.	Write a program to search the element using binary search.

7.	**Implement the following data structure techniques:**
a.	Write a program to create the tree and display the elements.
b.	Write a program to construct the binary tree.
c.	Write a program for inorder, postorder and preorder traversal of tree
8.	**Implement the following data structure techniques:**
a.	Write a program to insert the element into maximum heap.
b.	Write a program to insert the element into minimum heap.
9.	**Implement the following data structure techniques:**
a.	Write a program to implement the collision technique.
b.	Write a program to implement the concept of linear probing.
10.	**Implement the following data structure techniques:**
a.	Write a program to generate the adjacency matrix.
b.	Write a program for shortest path diagram.

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Data Structures and Algorithms Using Python	Rance Necaise	Wiley	First	2016
2.	Data Structures Using C and C++	Langsam , Augenstein, Tanenbaum	Pearson	First	2015

B. Sc. (Information Technology)		Semester – III	
Course Name: Computer Networks		**Course Code: USIT3P3**	
Periods per week (1 Period is 50 minutes)		**3**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Practical Examination**	2½	50
	Internal	--	--

List of Practical	
1.	**IPv4 Addressing and Subnetting** a) Given an IP address and network mask, determine other information about the IP address such as: • Network address • Network broadcast address • Total number of host bits • Number of hosts b) Given an IP address and network mask, determine other information about the IP address such as: • The subnet address of this subnet • The broadcast address of this subnet • The range of host addresses for this subnet • The maximum number of subnets for this subnet mask • The number of hosts for each subnet • The number of subnet bits • The number of this subnet
2.	Use of ping and tracert / traceroute, ipconfig / ifconfig, route and arp utilities.
3.	Configure IP static routing.
4.	Configure IP routing using RIP.
5.	Configuring Simple OSPF.
6.	Configuring DHCP server and client.
7.	Create virtual PC based network using virtualization software and virtual NIC.
8.	Configuring DNS Server and client.
9.	Configuring OSPF with multiple areas.
10.	Use of Wireshark to scan and check the packet information of following protocols • HTTP • ICMP • TCP • SMTP • POP3

B. Sc. (Information Technology)		Semester – III	
Course Name: Database Management System		Course Code: USIT3P4	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2½	50
	Internal	--	--

List of Practical	
1.	**SQL Statements – 1**
a.	Writing Basic SQL SELECT Statements
b.	Restricting and Sorting Data
c.	Single-Row Functions
2.	**SQL Statements – 2**
a.	Displaying Data from Multiple Tables
b.	Aggregating Data Using Group Functions
c.	Subqueries
3.	**Manipulating Data**
a.	Using INSERT statement
b.	Using DELETE statement
c.	Using UPDATE statement
4.	**Creating and Managing Tables**
a.	Creating and Managing Tables
b.	Including Constraints
5.	**Creating and Managing other database objects**
a.	Creating Views
b.	Other Database Objects
c.	Controlling User Access
6.	**Using SET operators, Date/Time Functions, GROUP BY clause (advanced features) and advanced subqueries**
a.	Using SET Operators
b.	Datetime Functions
c.	Enhancements to the GROUP BY Clause
d.	Advanced Subqueries
7.	**PL/SQL Basics**
a.	Declaring Variables
b.	Writing Executable Statements
c.	Interacting with the Oracle Server
d.	Writing Control Structures

8.	**Composite data types, cursors and exceptions.**
a.	Working with Composite Data Types
b.	Writing Explicit Cursors
c.	Handling Exceptions
9.	**Procedures and Functions**
a.	Creating Procedures
b.	Creating Functions
c.	Managing Subprograms
d.	Creating Packages
10.	**Creating Database Triggers**

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Database System and Concepts	A Silberschatz, H Korth, S Sudarshan	McGraw-Hill	Fifth Edition	
2.	Programming with PL/SQL for Beginners	H.Dand , R.Patil and T. Sambare	X –Team	First	2011
3.	PL/SQL Programming	Ivan Bayross	BPB	First	2010

B. Sc. (Information Technology)		Semester – III	
Course Name: Mobile Programming		Course Code: USIT3P5	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2½	50
	Internal	--	--

The practical's will be based on HTML5, CSS, CORDOVA and PhoneGAP API. (Android will be introduced later after they learn Java)

List of Practical	
	Setting up CORDOVA, PhoneGAP Project and environment.
1.	• Creating and building simple "Hello World" App using Cordova • Adding and Using Buttons • Adding and Using Event Listeners
2.	• Creating and Using Functions • Using Events • Handling and Using Back Button
3.	• Installing and Using Plugins • Installing and Using Battery Plugin • Installing and Using Camera Plugin
4.	• Installing and Using Contacts Plugin • Installing and Using Device Plugin • Installing and Using Accelerometer Plugin
5.	• Install and Using Device Orientation plugin • Install and Using Device Orientation plugin • Create and Using Prompt Function
6.	• Installing and Using File Plugin • Installing and Using File Transfer Plugin • Using Download and Upload functions
7.	• Installing and Using Globalization Plugin • Installing and Using Media Plugin • Installing and Using Media Capture Plugin
8.	• Installing and Using Network Information Plugin • Installing and Using Splash Screen Plugin • Installing and Using Vibration Plugin

9.	• Developing Single Page Apps
	• Developing Multipage Apps
	• Storing Data Locally in a Cordova App
10.	• Use of sqlite plugin with PhoneGap / apache Cordova
	• Using Sqlite read/write and search
	• Populating Cordova SQLite storage with the JQuery API

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Apache Cordova 4 Programming	John M. Wargo	Addison-Wesley Professional	1st	2015
2.	Apache Cordova in Action	Raymond Camden	Manning Publications	1st	2015
3.	PhoneGap By Example	Andrey Kovalenko	PACKT Publishing	1st	2015

SEMESTER IV

B. Sc. (Information Technology)		Semester – IV	
Course Name: Core Java		**Course Code: USIT401**	
Periods per week (1 Period is 50 minutes)		**5**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	2½	75
	Internal	--	25

Unit	Details	Lectures
I	**Introduction:** History, architecture and its components, Java Class File, Java Runtime Environment, The Java Virtual Machine, JVM Components, The Java API, java platform, java development kit, Lambda Expressions, Methods References, Type Annotations, Method Parameter Reflection, setting the path environment variable, Java Compiler And Interpreter, java programs, java applications, main(), public, static, void, string[] args, statements, white space, case sensitivity, identifiers, keywords, comments, braces and code blocks, variables, variable name **Data types:** primitive data types, Object Reference Types, Strings, Auto boxing, operators and properties of operators, Arithmetic operators, assignment operators, increment and decrement operator, relational operator, logical operator, bitwise operator, conditional operator.	12
II	**Control Flow Statements:** The If…Else If…Else Statement, The Switch…Case Statement **Iterations:** The While Loop, The Do … While Loop, The For Loop, The Foreach Loop, Labeled Statements, The Break And Continue Statements, The Return Statement **Classes:** Types of Classes, Scope Rules, Access Modifier, Instantiating Objects From A Class, Initializing The Class Object And Its Attributes, Class Methods, Accessing A Method, Method Returning A Value, Method's Arguments, Method Overloading, Variable Arguments [Varargs], Constructors, this Instance, super Instance, Characteristics Of Members Of A Class, constants, this instance, static fields of a class, static methods of a class, garbage collection.	12
III	**Inheritance:** Derived Class Objects, Inheritance and Access Control, Default Base Class Constructors, this and super keywords. Abstract Classes And Interfaces, Abstract Classes, Abstract Methods, Interfaces, What Is An Interface? How Is An Interface Different From An Abstract Class?, Multiple Inheritance, Default Implementation, Adding New Functionality, Method Implementation, Classes V/s Interfaces, Defining An Interface, Implementing Interfaces. **Packages:** Creating Packages, Default Package, Importing Packages, Using A Package.	12

IV	**Enumerations, Arrays:** Two Dimensional Arrays, Multi-Dimensional Arrays, Vectors, Adding Elements To A Vector, Accessing Vector Elements, Searching For Elements In A Vector, Working With The Size of The Vector. **Multithreading:** the thread control methods, thread life cycle, the main thread, creating a thread, extending the thread class. **Exceptions:** Catching Java Exceptions, Catching Run-Time Exceptions, Handling Multiple Exceptions, The finally Clause, The throws Clause **Byte streams:** reading console input, writing console output, reading file, writing file, writing binary data, reading binary data, getting started with character streams, writing file, reading file	**12**
V	**Event Handling:** Delegation Event Model, Events, Event classes, Event listener interfaces, Using delegation event model, adapter classes and inner classes. **Abstract Window Toolkit:** Window Fundamentals, Component, Container, Panel, Window, Frame, Canvas. Components – Labels, Buttons, Check Boxes, Radio Buttons, Choice Menus, Text Fields, Text, Scrolling List, Scrollbars, Panels, Frames **Layouts:** Flow Layout, Grid Layout, Border Layout, Card Layout.	**12**

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Core Java 8 for Beginners	Vaishali Shah, Sharnam Shah	SPD	1st	2015
2.	Java: The Complete Reference	Herbert Schildt	McGraw Hill	9th	2014
3.	Murach's beginning Java with Net Beans	Joel Murach , Michael Urban	SPD	1st	2016
4.	Core Java, Volume I: Fundamentals	Hortsman	Pearson	9th	2013
5.	Core Java, Volume II: Advanced Features	Gary Cornell and Hortsman	Pearson	8th	2008
6.	Core Java: An Integrated Approach	R. Nageswara Rao	DreamTech	1st	2008

B. Sc. (Information Technology)		Semester – IV	
Course Name: Introduction to Embedded Systems		**Course Code: USIT402**	
Periods per week (1 Period is 50 minutes)		**5**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	2½	75
	Internal	--	25

Unit	Details	Lectures
I	**Introduction:** Embedded Systems and general purpose computer systems, history, classifications, applications and purpose of embedded systems **Core of embedded systems:** microprocessors and microcontrollers, RISC and CISC controllers, Big endian and Little endian processors, Application specific ICs, Programmable logic devices, COTS, sensors and actuators, communication interface, embedded firmware, other system components. **Characteristics and quality attributes of embedded systems:** Characteristics, operational and non-operational quality attributes.	12
II	**Embedded Systems – Application and Domain Specific:** Application specific – washing machine, domain specific - automotive. **Embedded Hardware:** Memory map, i/o map, interrupt map, processor family, external peripherals, memory – RAM , ROM, types of RAM and ROM, memory testing, CRC ,Flash memory. **Peripherals:** Control and Status Registers, Device Driver, Timer Driver - Watchdog Timers.	12
III	**The 8051 Microcontrollers:** Microcontrollers and Embedded processors, Overview of 8051 family. 8051 Microcontroller hardware, Input/output pins, Ports, and Circuits, External Memory. **8051 Programming in C:** Data Types and time delay in 8051 C, I/O Programming, Logic operations, Data conversion Programs.	12
IV	**Designing Embedded System with 8051 Microcontroller:** Factors to be considered in selecting a controller, why 8051 Microcontroller, Designing with 8051. **Programming embedded systems:** structure of embedded program, infinite loop, compiling, linking and debugging.	12
V	**Real Time Operating System (RTOS):** Operating system basics, types of operating systems, Real-Time Characteristics, Selection Process of an RTOS. **Design and Development:** Embedded system development Environment – IDE, types of file generated on cross compilation, disassembler/ de-compiler, simulator, emulator and debugging, embedded product development life-cycle, trends in embedded industry.	12

Books and References:					
Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Programming Embedded Systems in C and C++	Michael Barr	O'Reilly	First	1999
2.	Introduction to embedded systems	Shibu K V	Tata Mcgraw-Hill	First	2012
3.	The 8051 Microcontroller and Embedded Systems	Muhammad Ali Mazidi	Pearson	Second	2011
4.	Embedded Systems	Rajkamal	Tata Mcgraw-Hill		

B. Sc. (Information Technology)		Semester – IV	
Course Name: Computer Oriented Statistical Techniques		**Course Code: USIT403**	
Periods per week (1 Period is 50 minutes)		**5**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	**2½**	**75**
	Internal	**--**	**25**

Unit	Details	Lectures
I	**The Mean, Median, Mode, and Other Measures of Central Tendency**: Index, or Subscript, Notation, Summation Notation, Averages, or Measures of Central Tendency ,The Arithmetic Mean , The Weighted Arithmetic Mean ,Properties of the Arithmetic Mean ,The Arithmetic Mean Computed from Grouped Data ,The Median ,The Mode, The Empirical Relation Between the Mean, Median, and Mode, The Geometric Mean G, The Harmonic Mean H ,The Relation Between the Arithmetic, Geometric, and Harmonic Means, The Root Mean Square, Quartiles, Deciles, and Percentiles, Software and Measures of Central Tendency. **The Standard Deviation and Other Measures of Dispersion**: Dispersion, or Variation, The Range, The Mean Deviation, The Semi-Interquartile Range, The 10–90 Percentile Range, The Standard Deviation, The Variance, Short Methods for Computing the Standard Deviation, Properties of the Standard Deviation, Charlie's Check, Sheppard's Correction for Variance, Empirical Relations Between Measures of Dispersion, Absolute and Relative Dispersion; Coefficient of Variation, Standardized Variable; Standard Scores, Software and Measures of Dispersion. **Introduction to R:** Basic syntax, data types, variables, operators, control statements, R-functions, R –Vectors, R – lists, R Arrays.	**12**
II	**Moments, Skewness, and Kurtosis** : Moments , Moments for Grouped Data ,Relations Between Moments , Computation of Moments for Grouped Data, Charlie's Check and Sheppard's Corrections, Moments in Dimensionless Form, Skewness, Kurtosis, Population Moments, Skewness, and Kurtosis, Software Computation of Skewness and Kurtosis. **Elementary Probability Theory**: Definitions of Probability, Conditional Probability; Independent and Dependent Events, Mutually Exclusive Events, Probability Distributions, Mathematical Expectation, Relation Between Population, Sample Mean, and Variance, Combinatorial Analysis, Combinations, Stirling's Approximation to n!, Relation of Probability to Point Set Theory, Euler or Venn Diagrams and Probability. **Elementary Sampling Theory** : Sampling Theory, Random Samples and Random Numbers, Sampling With and Without Replacement,	**12**

	Sampling Distributions, Sampling Distribution of Means, Sampling Distribution of Proportions, Sampling Distributions of Differences and Sums, Standard Errors, Software Demonstration of Elementary Sampling Theory.		
III	**Statistical Estimation Theory**: Estimation of Parameters, Unbiased Estimates, Efficient Estimates, Point Estimates and Interval Estimates; Their Reliability, Confidence-Interval Estimates of Population Parameters, Probable Error. **Statistical Decision Theory:** Statistical Decisions, Statistical Hypotheses, Tests of Hypotheses and Significance, or Decision Rules, Type I and Type II Errors, Level of Significance, Tests Involving Normal Distributions, Two-Tailed and One-Tailed Tests, Special Tests, Operating-Characteristic Curves; the Power of a Test, p-Values for Hypotheses Tests, Control Charts, Tests Involving Sample Differences, Tests Involving Binomial Distributions. **Statistics in R:** mean, median, mode, Normal Distribution , Binomial Distribution, Frequency Distribution in R.	**12**	
IV	**Small Sampling Theory**: Small Samples, Student's t Distribution, Confidence Intervals, Tests of Hypotheses and Significance, The Chi-Square Distribution, Confidence Intervals for Sigma , Degrees of Freedom, The F Distribution. **The Chi-Square Test**: Observed and Theoretical Frequencies, Definition of chi-square, Significance Tests, The Chi-Square Test for Goodness of Fit, Contingency Tables, Yates' Correction for Continuity, Simple Formulas for Computing chi-square, Coefficient of Contingency, Correlation of Attributes, Additive Property of chi-square.	**12**	
V	**Curve Fitting and the Method of Least Squares:** Relationship Between Variables, Curve Fitting, Equations of Approximating Curves, Freehand Method of Curve Fitting, The Straight Line, The Method of Least Squares, The Least-Squares Line, Nonlinear Relationships, The Least-Squares Parabola, Regression, Applications to Time Series, Problems Involving More Than Two Variables. **Correlation Theory:** Correlation and Regression, Linear Correlation, Measures of Correlation, The Least-Squares Regression Lines, Standard Error of Estimate, Explained and Unexplained Variation, Coefficient of Correlation, Remarks Concerning the Correlation Coefficient, Product-Moment Formula for the Linear Correlation Coefficient, Short Computational Formulas, Regression Lines and the Linear Correlation Coefficient, Correlation of Time Series, Correlation of Attributes, Sampling Theory of Correlation, Sampling Theory of Regression.	**12**	

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	STATISTICS	Murray R. Spiegel, Larry J. Stephens.	McGRAW – HILL ITERNATIONAL	FOURTH	
2.	A Practical Approach using R	R.B. Patil, H.J. Dand and R. Bhavsar	SPD	1st	2017
3.	FUNDAMENTAL OF MATHEMATICAL STATISTICS	S.C. GUPTA and V.K. KAPOOR	SULTAN CHAND and SONS	ELEVENTH REVISED	2011
4.	MATHEMATICAL STATISTICS	J.N. KAPUR and H.C. SAXENA	S. CHAND	TWENTIETH REVISED	2005

B. Sc. (Information Technology)		Semester – IV	
Course Name: Software Engineering		**Course Code: USIT404**	
Periods per week (1 Period is 50 minutes)		**5**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	2½	75
	Internal	--	25

Unit	Details	Lectures
I	**Introduction:** What is software engineering? Software Development Life Cycle, Requirements Analysis, Software Design, Coding, Testing, Maintenance etc. **Software Requirements:** Functional and Non-functional requirements, User Requirements, System Requirements, Interface Specification, Documentation of the software requirements. **Software Processes:** Process and Project, Component Software Processes. **Software Development Process Models.** • Waterfall Model. • Prototyping. • Iterative Development. • Rational Unified Process. • The RAD Model • Time boxing Model. **Agile software development**: Agile methods, Plan-driven and agile development, Extreme programming, Agile project management, Scaling agile methods.	**12**
II	**Socio-technical system:** Essential characteristics of socio technical systems, Emergent System Properties, Systems Engineering, Components of system such as organization, people and computers, Dealing Legacy Systems. **Critical system:** Types of critical system, A simple safety critical system, Dependability of a system, Availability and Reliability, Safety and Security of Software systems. **Requirements Engineering Processes:** Feasibility study, Requirements elicitation and analysis, Requirements Validations, Requirements Management. **System Models:** Models and its types, Context Models, Behavioural Models, Data Models, Object Models, Structured Methods.	**12**
III	**Architectural Design:** Architectural Design Decisions, System Organisation, Modular Decomposition Styles, Control Styles, Reference Architectures.	**12**

	User Interface Design: Need of UI design, Design issues, The UI design Process, User analysis, User Interface Prototyping, Interface Evaluation. **Project Management** Software Project Management, Management activities, Project Planning, Project Scheduling, Risk Management. **Quality Management:** Process and Product Quality, Quality assurance and Standards, Quality Planning, Quality Control, Software Measurement and Metrics.	
IV	**Verification and Validation:** Planning Verification and Validation, Software Inspections, Automated Static Analysis, Verification and Formal Methods. **Software Testing:** System Testing, Component Testing, Test Case Design, Test Automation. **Software Measurement**: Size-Oriented Metrics, Function-Oriented Metrics, Extended Function Point Metrics **Software Cost Estimation:** Software Productivity, Estimation Techniques, Algorithmic Cost Modelling, Project Duration and Staffing	**12**
V	**Process Improvement:** Process and product quality, Process Classification, Process Measurement, Process Analysis and Modeling, Process Change, The CMMI Process Improvement Framework. **Service Oriented Software Engineering:** Services as reusable components, Service Engineering, Software Development with Services. **Software reuse:** The reuse landscape, Application frameworks, Software product lines, COTS product reuse. **Distributed software engineering**: Distributed systems issues, Client–server computing, Architectural patterns for distributed systems, Software as a service	**12**

Books and References:					
Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Software Engineering, edition,	Ian Somerville	Pearson Education.	Ninth	
2.	Software Engineering	Pankaj Jalote	Narosa Publication		
3.	Software engineering, a practitioner's approach	Roger Pressman	Tata Mcgraw-hill	Seventh	
4.	Software Engineering principles and practice	WS Jawadekar	Tata Mcgraw-hill		
5.	Software Engineering-A Concise Study	S.A Kelkar	PHI India.		

6.	Software Engineering Concept and Applications	Subhajit Datta	Oxford Higher Education		
7.	Software Design	D.Budgen	Pearson education	2nd	
8.	Software Engineering	KL James	PHI	EEE	2009

B. Sc. (Information Technology)	Semester – IV
Course Name: Computer Graphics and Animation	Course Code: USIT405
Periods per week (1 Period is 50 minutes)	5
Credits	2

		Hours	Marks
Evaluation System	Theory Examination	2½	75
	Internal	--	25

Unit	Details	Lectures
I	**Introduction to Computer Graphics:** Overview of Computer Graphics, Computer Graphics Application and Software, Description of some graphics devices, Input Devices for Operator Interaction, Active and Passive Graphics Devices, Display Technologies, Storage Tube Graphics Displays, Calligraphic Refresh Graphics Displays, Raster Refresh (Raster-Scan) Graphics Displays, Cathode Ray Tube Basics, Color CRT Raster Scan Basics, Video Basics, The Video Controller, Random-Scan Display Processor, LCD displays. **Scan conversion** – Digital Differential Analyzer (DDA) algorithm, Bresenhams' Line drawing algorithm. Bresenhams' method of Circle drawing, Midpoint Circle Algorithm, Midpoint Ellipse Algorithm, Mid-point criteria, Problems of Aliasing, end-point ordering and clipping lines, Scan Converting Circles, Clipping Lines algorithms– Cyrus-Beck, Cohen-Sutherland and Liang-Barsky, Clipping Polygons, problem with multiple components.	12
II	**Two-Dimensional Transformations:** Transformations and Matrices, Transformation Conventions, 2D Transformations, Homogeneous Coordinates and Matrix Representation of 2D Transformations, Translations and Homogeneous Coordinates, Rotation, Reflection, Scaling, Combined Transformation, Transformation of Points, Transformation of The Unit Square, Solid Body Transformations, Rotation About an Arbitrary Point, Reflection through an Arbitrary Line, A Geometric Interpretation of Homogeneous Coordinates, The Window-to-Viewport Transformations. **Three-Dimensional Transformations:** Three-Dimensional Scaling, Three-Dimensional Shearing, Three-Dimensional Rotation, Three-Dimensional Reflection, Three-Dimensional Translation, Multiple Transformation, Rotation about an Arbitrary Axis in Space, Reflection through an Arbitrary Plane, Matrix Representation of 3D Transformations, Composition of 3D Transformations, Affine and Perspective Geometry, Perspective Transformations, Techniques for Generating Perspective Views, Vanishing Points, the Perspective Geometry and camera models, Orthographic Projections, Axonometric Projections, Oblique Projections, View volumes for projections.	12

III	**Viewing in 3D** Stages in 3D viewing, Canonical View Volume (CVV), Specifying an Arbitrary 3D View, Examples of 3D Viewing, The Mathematics of Planar Geometric Projections, Combined transformation matrices for projections and viewing, Coordinate Systems and matrices, camera model and viewing pyramid. **Light:** Radiometry, Transport, Equation, Photometry **Color:** Colorimetry, Color Spaces, Chromatic Adaptation, Color Appearance	**12**
IV	**Visible-Surface Determination:** Techniques for efficient Visible-Surface Algorithms, Categories of algorithms, Back face removal, The z-Buffer Algorithm, Scan-line method, Painter's algorithms (depth sorting), Area sub-division method, BSP trees, Visible-Surface Ray Tracing, comparison of the methods. **Plane Curves and Surfaces:** Curve Representation, Nonparametric Curves, Parametric Curves, Parametric Representation of a Circle, Parametric Representation of an Ellipse, Parametric Representation of a Parabola, Parametric Representation of a Hyperbola, Representation of Space Curves, Cubic Splines, , Bezier Curves, B-spline Curves, B-spline Curve Fit, B-spline Curve Subdivision, Parametric Cubic Curves, Quadric Surfaces. Bezier Surfaces.	**12**
V	**Computer Animation:** Principles of Animation, Key framing, Deformations, Character Animation, Physics-Based Animation, Procedural Techniques, Groups of Objects. **Image Manipulation and Storage:** What is an Image? Digital image file formats, Image compression standard – JPEG, Image Processing - Digital image enhancement, contrast stretching, Histogram Equalization, smoothing and median Filtering.	**12**

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Computer Graphics - Principles and Practice	J. D. Foley, A. Van Dam, S. K. Feiner and J. F. Hughes	Pearson	2nd	
2.	Steve Marschner, Peter Shirley	Fundamentals of Computer Graphics	CRC press	4th	2016
3.	Computer Graphics	Hearn, Baker	Pearson	2nd	
4.	Principles of Interactive Computer Graphics	William M. Newman and Robert F. Sproull	TMH	2nd	
5.	Mathematical Elements for CG	D. F. Rogers, J. A. Adams	TMH	2nd	

B. Sc. (Information Technology)		Semester –IV	
Course Name: Core Java Practical		**Course Code: USIT4P1**	
Periods per week **1 Period is 50 minutes**	**Lectures per week**	**3**	
		Hours	**Marks**
Evaluation System	**Practical Examination**	**2½**	**50**

List of Practical	
1.	**Java Basics**
a.	Write a Java program that takes a number as input and prints its multiplication table upto 10.

b.	Write a Java program to display the following pattern.
	*****
	 ****
	  ***
	  **
	   *

c.	Write a Java program to print the area and perimeter of a circle.
2.	**Use of Operators**
a.	Write a Java program to add two binary numbers.
b.	Write a Java program to convert a decimal number to binary number and vice versa.
c.	Write a Java program to reverse a string.
3.	**Java Data Types**
a.	Write a Java program to count the letters, spaces, numbers and other characters of an input string.
b.	Implement a Java function that calculates the sum of digits for a given char array consisting of the digits '0' to '9'. The function should return the digit sum as a long value.
c.	Find the smallest and largest element from the array
4.	**Methods and Constructors**
a.	Designed a class SortData that contains the method asec() and desc().
b.	Designed a class that demonstrates the use of constructor and destructor.
c.	Write a java program to demonstrate the implementation of abstract class.
5.	**Inheritance**
a.	Write a java program to implement single level inheritance.
b.	Write a java program to implement method overriding
c.	Write a java program to implement multiple inheritance.

6.	**Packages and Arrays**
a.	Create a package, Add the necessary classes and import the package in java class.
b.	Write a java program to add two matrices and print the resultant matrix.
c.	Write a java program for multiplying two matrices and print the product for the same.
7.	**Vectors and Multithreading**
a.	Write a java program to implement the vectors.
b.	Write a java program to implement thread life cycle.
c.	Write a java program to implement multithreading.
8.	**File Handling**
a.	Write a java program to open a file and display the contents in the console window.
b.	Write a java program to copy the contents from one file to other file.
c.	Write a java program to read the student data from user and store it in the file.
9.	**GUI and Exception Handling**
a.	Design a AWT program to print the factorial for an input value.
b.	Design an AWT program to perform various string operations like reverse string, string concatenation etc.
c.	Write a java program to implement exception handling.
10.	**GUI Programming.**
a.	Design an AWT application that contains the interface to add student information and display the same.
b.	Design a calculator based on AWT application.
c.	Design an AWT application to generate result marks sheet.

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Core Java 8 for Beginners	Vaishali Shah, Sharnam Shah	SPD	1st	2015
2.	Java: The Complete Reference	Herbert Schildt	McGraw Hill	9th	2014
3.	Murach's beginning Java with Net Beans	Joel Murach , Michael Urban	SPD	1st	2016
4.	Core Java, Volume I: Fundamentals	Hortsman	Pearson	9th	2013
5.	Core Java, Volume II: Advanced Features	Gary Cornell and Hortsman	Pearson	8th	2008
6.	Core Java: An Integrated Approach	R. Nageswara Rao	DreamTech	1st	2008

B. Sc. (Information Technology)		Semester – IV	
Course Name: Introduction to Embedded Systems Practical		Course Code: USIT4P2	
Periods per week 1 Period is 50 minutes	Lectures per week	3	
		Hours	Marks
Evaluation System	Practical Examination	2½	50

List of Practical	

1.	Design and develop a reprogrammable embedded computer using 8051 microcontrollers and to show the following aspects.
	    a. Programming
	    b. Execution
	    c. Debugging

2. A	Configure timer control registers of 8051 and develop a program to generate given time delay.
B	To demonstrate use of general purpose port i.e. Input/ output port of two controllers for data transfer between them.
3. A	Port I / O: Use one of the four ports of 8051 for O/P interfaced to eight LED's. Simulate binary counter (8 bit) on LED's
B	To interface 8 LEDs at Input-output port and create different patterns.
C	To demonstrate timer working in timer mode and blink LED without using any loop delay routine.
4. A	Serial I / O: Configure 8051 serial port for asynchronous serial communication with serial port of PC exchange text messages to PC and display on PC screen. Signify end of message by carriage return.
B	To demonstrate interfacing of seven-segment LED display and generate counting from 0 to 99 with fixed time delay.
C	Interface 8051 with D/A converter and generate square wave of given frequency on oscilloscope.
5. A	Interface 8051 with D/A converter and generate triangular wave of given frequency on oscilloscope.
B	Using D/A converter generate sine wave on oscilloscope with the help of lookup table stored in data area of 8051.
6.	Interface stepper motor with 8051 and write a program to move the motor through a given angle in clock wise or counter clock wise direction.

7.	Generate traffic signal.
8.	Implement Temperature controller.
9.	Implement Elevator control.
10.	**Using FlashMagic**
A	To demonstrate the procedure for flash programming for reprogrammable embedded system board using FlashMagic
B	To demonstrate the procedure and connections for multiple controllers programming of same type of controller with same source code in one go, using flash magic.

B. Sc. (Information Technology)		Semester – IV	
Course Name: Computer Oriented Statistical Techniques Practical		Course Code: USIT4P3	
Periods per week 1 Period is 50 minutes	Lectures per week	3	
		Hours	Marks
Evaluation System	Practical Examination	2½	50

List of Practical	
1.	Using R execute the basic commands, array, list and frames.
2.	Create a Matrix using R and Perform the operations addition, inverse, transpose and multiplication operations.
3.	Using R Execute the statistical functions: mean, median, mode, quartiles, range, inter quartile range histogram
4.	Using R import the data from Excel / .CSV file and Perform the above functions.
5.	Using R import the data from Excel / .CSV file and Calculate the standard deviation, variance, co-variance.
6.	Using R import the data from Excel / .CSV file and draw the skewness.
7.	Import the data from Excel / .CSV and perform the hypothetical testing.
8.	Import the data from Excel / .CSV and perform the Chi-squared Test.
9.	Using R perform the binomial and normal distribution on the data.
10.	Perform the Linear Regression using R.
11.	Compute the Least squares means using R.
12.	Compute the Linear Least Square Regression

Books and References:					
Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	A Practical Approach to R Tool	R.B. Patil, H.J. Dand and R. Dahake	SPD	First	2011
2.	STATISTICS	Murray R. Spiegel, Larry J. Stephens.	McGRAW –HILL INTERNATIONAL	FOURTH	2006

B. Sc. (Information Technology)		Semester – IV	
Course Name: Software Engineering		Course Code: USIT4P4	
Periods per week 1 Period is 50 minutes	Lectures per week	3	
		Hours	Marks
Evaluation System	Practical Examination	2½	50

List of Practical (To be executed using Star UML or any similar software)	
1.	Study and implementation of class diagrams.
2.	Study and implementation of Use Case Diagrams.
3.	Study and implementation of Entity Relationship Diagrams.
4.	Study and implementation of Sequence Diagrams.
5.	Study and implementation of State Transition Diagrams.
6.	Study and implementation of Data Flow Diagrams.
7.	Study and implementation of Collaboration Diagrams.
8.	Study and implementation of Activity Diagrams.
9.	Study and implementation of Component Diagrams.
10.	Study and implementation of Deployment Diagrams.

Books and References:					
Sr. No.	Title	Author/s	Publisher	Edition	Year
3.	Object - Oriented Modeling and Design	Michael Blaha, James Rumbaugh	Pearson		2011
4.	Learning UML 2. 0	Kim Hamilton, Russ Miles	O'Reilly Media		2006
5.	The unified modeling language user guide	Grady Booch, James Rumbaugh, Ivar Jacobson	Addison-Wesley		2005
6.	UML A Beginners Guide	Jason T. Roff	McGraw Hill Professional		2003

B. Sc. (Information Technology)		Semester – IV	
Course Name: Computer Graphics and Animation		**Course Code: USIT4P5**	
Periods per week **1 Period is 50 minutes**	**Lectures per week**	**3**	
		Hours	**Marks**
Evaluation System	**Practical Examination**	**2½**	**50**

List of Practical	
1.	**Solve the following:**
a.	Study and enlist the basic functions used for graphics in C / C++ / Python language. Give an example for each of them.
b.	Draw a co-ordinate axis at the center of the screen.
2.	**Solve the following:**
a.	Divide your screen into four region, draw circle, rectangle, ellipse and half ellipse in each region with appropriate message.
b.	Draw a simple hut on the screen.
3.	**Draw the following basic shapes in the center of the screen :**
	i. Circle ii. Rectangle iii. Square iv. Concentric Circles v. Ellipse vi. Line
4.	**Solve the following:**
a.	Develop the program for DDA Line drawing algorithm.
b.	Develop the program for Bresenham's Line drawing algorithm.
5.	**Solve the following:**
a.	Develop the program for the mid-point circle drawing algorithm.
b.	Develop the program for the mid-point ellipse drawing algorithm.
6.	**Solve the following:**
a.	Write a program to implement 2D scaling.
b.	Write a program to perform 2D translation
7.	**Solve the following:**
a.	Perform 2D Rotation on a given object.
b.	Program to create a house like figure and perform the following operations. i. Scaling about the origin followed by translation. ii. Scaling with reference to an arbitrary point. iii. Reflect about the line y = mx + c.

8.	Solve the following:
a.	Write a program to implement Cohen-Sutherland clipping.
b.	Write a program to implement Liang - Barsky Line Clipping Algorithm
9.	Solve the following:
a.	Write a program to fill a circle using Flood Fill Algorithm.
b.	Write a program to fill a circle using Boundary Fill Algorithm.
10.	Solve the following:
a.	Develop a simple text screen saver using graphics functions.
b.	Perform smiling face animation using graphic functions.
c.	Draw the moving car on the screen.

Books and References:					
Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Computer Graphics - Principles and Practice	J. D. Foley, A. Van Dam, S. K. Feiner and J. F. Hughes	Pearson Education	Second Edition	
2.	Steve Marschner, Peter Shirley	Fundamentals of Computer Graphics	CRC press	Fourth Edition	2016
3.	Computer Graphics	Hearn, Baker	Pearson Education	Second	
4.	Principles of Interactive Computer Graphics	William M. Newman and Robert F. Sproull	Tata McGraw Hill	Second	

UNIVERSITY OF MUMBAI

Syllabus for T.Y.B.Sc.

Programme: B.Sc.

Subject: Information Technology

with effect from the academic year

2018 – 2019

Semester – 5			
Course Code	**Course Type**	**Course Title**	**Credits**
USIT501	Skill Enhancement Course	Software Project Management	2
USIT502	Skill Enhancement Course	Internet of Things	2
USIT503	Skill Enhancement Course	Advanced Web Programming	2
USIT504	Discipline Specific Elective (Any One)	Artificial Intelligence	2
USIT505		Linux System Administration	
USIT506	Discipline Specific Elective (Any One)	Enterprise Java	2
USIT507		Next Generation Technologies	
USIT5P1	Skill Enhancement Course Practical	Project Dissertation	2
USIT5P2	Skill Enhancement Course Practical	Internet of Things Practical	2
USIT5P3	Skill Enhancement Course Practical	Advanced Web Programming Practical	2
USIT5P4	Discipline Specific Elective Practical (Any One)*	Artificial Intelligence Practical	2
USIT5P5		Linux Administration Practical	
USIT5P6	Discipline Specific Elective Practical (Any One)*	Enterprise Java Practical	2
USIT5P7		Next Generation Technologies Practical	
		Total Credits	**20**

(All the practical mentioned in the syllabi are compulsory as per the courses chosen)

Semester – 6			
Course Code	**Course Type**	**Course Title**	**Credits**
USIT601	Skill Enhancement Course	Software Quality Assurance	2
USIT602	Skill Enhancement Course	Security in Computing	2
USIT603	Skill Enhancement Course	Business Intelligence	2
USIT604	Discipline Specific Elective (Any One)	Principles of Geographic Information Systems	2
USIT605		Enterprise Networking	
USIT606	Discipline Specific Elective (Any One)	IT Service Management	2
USIT607		Cyber Laws	
USIT6P1	Skill Enhancement Course Practical	Project Implementation	2
USIT6P2	Skill Enhancement Course Practical	Security in Computing Practical	2
USIT6P3	Skill Enhancement Course Practical	Business Intelligence Practical	2
USIT6P4	Discipline Specific Elective Practical (Any One)*	Principles of Geographic Information Systems Practical	2
USIT6P5		Enterprise Networking Practical	
USIT6P6	Skill Enhancement Course Practical	Advanced Mobile Programming	2
		Total Credits	**20**

*The choice of Practical course is based on the theory Course. For Semester V, USIT504, USIT505, USIT506 and USIT507, the practical courses are USIT5P4, USIT5P5 USIT5P6, USIT5P7. For Semester VI, USIT604, USIT605 the practical courses are USIT6P4, USIT6P5 respectively. Practical Course USIT6P6 is compulsory.

SEMESTER V

B. Sc. (Information Technology)		Semester – V	
Course Name: Software Project Management		**Course Code: USIT501**	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	**Marks**
Evaluation System	Theory Examination	2½	75
	Internal	--	25

Unit	Details	Lectures
I	**Introduction to Software Project Management**: Introduction, Why is Software Project Management Important? What is a Project? Software Projects versus Other Types of Project, Contract Management and Technical Project Management, Activities Covered by Software Project Management, Plans, Methods and Methodologies, Some Ways of Categorizing Software Projects, Project Charter, Stakeholders, Setting Objectives, The Business Case, Project Success and Failure, What is Management? Management Control, Project Management Life Cycle, Traditional versus Modern Project Management Practices. **Project Evaluation and Programme Management**: Introduction, Business Case, Project Portfolio Management, Evaluation of Individual Projects, Cost–benefit Evaluation Techniques, Risk Evaluation, Programme Management, Managing the Allocation of Resources within Programmes, Strategic Programme Management, Creating a Programme, Aids to Programme Management, Some Reservations about Programme Management, Benefits Management. **An Overview of Project Planning** : Introduction to Step Wise Project Planning, Step 0: Select Project, Step 1: Identify Project Scope and Objectives, Step 2: Identify Project Infrastructure, Step 3: Analyse Project Characteristics, Step 4: Identify Project Products and Activities, Step 5: Estimate Effort for Each Activity, Step 6: Identify Activity Risks, Step 7: Allocate Resources, Step 8: Review/Publicize Plan, Steps 9 and 10: Execute Plan/Lower Levels of Planning	**12**
II	**Selection of an Appropriate Project Approach:** Introduction, Build or Buy? Choosing Methodologies and Technologies, Software Processes and Process Models, Choice of Process Models, Structure versus Speed of Delivery, The Waterfall Model, The Spiral Model, Software Prototyping, Other Ways of Categorizing Prototypes, Incremental Delivery, Atern/Dynamic Systems Development Method, Rapid Application Development, Agile Methods, Extreme Programming (XP), Scrum, Lean Software Development, Managing Iterative Processes, Selecting the Most Appropriate Process Model. **Software Effort Estimation:** Introduction, Where are the Estimates Done? Problems with Over- and Under-Estimates, The Basis for Software Estimating, Software Effort Estimation Techniques, Bottom-up Estimating, The Top-down Approach and Parametric Models, Expert Judgement, Estimating by Analogy, Albrecht Function Point	**12**

	Analysis, Function Points Mark II, COSMIC Full Function Points, COCOMO II: A Parametric Productivity Model, Cost Estimation, Staffing Pattern, Effect of Schedule Compression, Capers Jones Estimating Rules of Thumb.	
III	**Activity Planning:** Introduction, Objectives of Activity Planning, When to Plan, Project Schedules, Projects and Activities, Sequencing and Scheduling Activities, Network Planning Models, Formulating a Network Model, Adding the Time Dimension, The Forward Pass, Backward Pass, Identifying the Critical Path, Activity Float, Shortening the Project Duration, Identifying Critical Activities, Activity-on-Arrow Networks. **Risk Management**: Introduction, Risk, Categories of Risk, Risk Management Approaches, A Framework for Dealing with Risk, Risk Identification, Risk Assessment, Risk Planning, Risk Management, Evaluating Risks to the Schedule, Boehm's Top 10 Risks and Counter Measures, Applying the PERT Technique, Monte Carlo Simulation, Critical Chain Concepts. **Resource Allocation**: Introduction, Nature of Resources, Identifying Resource Requirements, Scheduling Resources, Creating Critical Paths, Counting the Cost, Being Specific, Publishing the Resource Schedule, Cost Schedules, Scheduling Sequence.	**12**
IV	**Monitoring and Control**: Introduction, Creating the Framework, Collecting the Data, Review, Visualizing Progress, Cost Monitoring, Earned Value Analysis, Prioritizing Monitoring, Getting the Project Back to Target, Change Control, Software Configuration Management (SCM). **Managing Contracts**: Introduction, Types of Contract, Stages in Contract Placement, Typical Terms of a Contract, Contract Management, Acceptance. **Managing People in Software Environments**: Introduction, Understanding Behaviour, Organizational Behaviour: A Background, Selecting the Right Person for the Job, Instruction in the Best Methods, Motivation, The Oldham–Hackman Job Characteristics Model, Stress, Stress Management, Health and Safety, Some Ethical and Professional Concerns.	**12**
V	**Working in Teams:** Introduction, Becoming a Team, Decision Making, Organization and Team Structures, Coordination Dependencies, Dispersed and Virtual Teams, Communication Genres, Communication Plans, Leadership. **Software Quality :** Introduction, The Place of Software Quality in Project Planning, Importance of Software Quality, Defining Software Quality, Software Quality Models, ISO 9126, Product and Process Metrics, Product versus Process Quality Management, Quality Management Systems, Process Capability Models, Techniques to Help Enhance Software Quality, Testing, Software Reliability, Quality Plans.	**12**

| | **Project Closeout**: Introduction, Reasons for Project Closure, Project Closure Process, Performing a Financial Closure, Project Closeout Report. | |

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Software Project Management	Bob Hughes, Mike Cotterell, Rajib Mall	TMH	6th	2018
2.	Project Management and Tools & Technologies – An overview	Shailesh Mehta	SPD	1st	2017
3.	Software Project Management	Walker Royce	Pearson		2005

B. Sc. (Information Technology)		Semester – V	
Course Name: Internet of Things		**Course Code: USIT502**	
Periods per week (1 Period is 50 minutes)		**5**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	2½	75
	Internal	--	25

Unit	Details	Lectures
I	**The Internet of Things: An Overview** : The Flavour of the Internet of Things, The "Internet" of "Things", The Technology of the Internet of Things, Enchanted Objects, Who is Making the Internet of Things? **Design Principles for Connected Devices:** Calm and Ambient Technology, Magic as Metaphor, Privacy, Keeping Secrets, Whose Data Is It Anyway? Web Thinking for Connected Devices, Small Pieces, Loosely Joined, First-Class Citizens On The Internet, Graceful Degradation, Affordances. **Internet Principles:** Internet Communications: An Overview, IP, TCP, The IP Protocol Suite (TCP/IP), UDP, IP Addresses, DNS, Static IP Address Assignment, Dynamic IP Address Assignment, IPv6, MAC Addresses, TCP and UDP Ports, An Example: HTTP Ports, Other Common Ports, Application Layer Protocols, HTTP, HTTPS: Encrypted HTTP, Other Application Layer Protocols.	12
II	**Thinking About Prototyping:** Sketching, Familiarity, Costs versus Ease of Prototyping, Prototypes and Production, Changing Embedded Platform, Physical Prototypes and Mass Personalisation, Climbing into the Cloud, Open Source versus Closed Source, Why Closed? Why Open? Mixing Open and Closed Source, Closed Source for Mass Market Projects, Tapping into the Community. **Prototyping Embedded Devices:** Electronics, Sensors, Actuators, Scaling Up the Electronics, Embedded Computing Basics, Microcontrollers, System-on-Chips, Choosing Your Platform, Arduino, Developing on the Arduino, Some Notes on the Hardware, Openness, Raspberry Pi, Cases and Extension Boards, Developing on the Raspberry Pi, Some Notes on the Hardware, Openness.	12
III	**Prototyping the Physical Design:** Preparation, Sketch, Iterate, and Explore, Nondigital Methods, Laser Cutting, Choosing a Laser Cutter, Software, Hinges and Joints, 3D Printing, Types of 3D Printing, Software, CNC Milling, Repurposing/Recycling. **Prototyping Online Components:** Getting Started with an API, Mashing Up APIs, Scraping, Legalities, Writing a New API, Clockodillo, Security, Implementing the API, Using Curl to Test, Going Further, Real-Time Reactions, Polling, Comet, Other Protocols, MQ Telemetry Transport,	12

8

		Extensible Messaging and Presence Protocol, Constrained Application Protocol.	
IV		**Techniques for Writing Embedded Code:** Memory Management, Types of Memory, Making the Most of Your RAM, Performance and Battery Life, Libraries, Debugging, **Business Models:** A Short History of Business Models, Space and Time, From Craft to Mass Production, The Long Tail of the Internet, Learning from History, The Business Model Canvas, Who Is the Business Model For? Models, Make Thing, Sell Thing, Subscriptions, Customisation, Be a Key Resource, Provide Infrastructure: Sensor Networks, Take a Percentage, Funding an Internet of Things Startup, Hobby Projects and Open Source, Venture Capital, Government Funding, Crowdfunding, Lean Startups.	**12**
V		**Moving to Manufacture:** What Are You Producing? Designing Kits, Designing Printed circuit boards, Software Choices, The Design Process, Manufacturing Printed Circuit Boards, Etching Boards, Milling Boards. Assembly, Testing, Mass-Producing the Case and Other Fixtures, Certification, Costs, Scaling Up Software, Deployment, Correctness and Maintainability, Security, Performance, User Community. **Ethics:** Characterizing the Internet of Things, Privacy, Control, Disrupting Control, Crowdsourcing, Environment, Physical Thing, Electronics, Internet Service, Solutions, The Internet of Things as Part of the Solution, Cautious Optimism, The Open Internet of Things Definition.	**12**

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Designing the Internet of Things	Adrian McEwen, Hakim Cassimally	WILEY	First	2014
2.	Internet of Things – Architecture and Design	Raj Kamal	McGraw Hill	First	2017
3.	Getting Started with the Internet of Things	Cuno Pfister	O'Reilly	Sixth	2018
4.	Getting Started with Raspberry Pi	Matt Richardson and Shawn Wallace	SPD	Third	2016

B. Sc. (Information Technology)		Semester – V	
Course Name: Advanced Web Programming		**Course Code: USIT503**	
Periods per week (1 Period is 50 minutes)		**5**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	2½	75
	Internal	--	25

Unit	Details	Lectures
I	**Introducing .NET**: The .NET Framework, C#, VB, and the .NET Languages, The Common Language Runtime, The .NET Class Library. **The C# Language**: C# Language Basics, Variables and Data Types, Variable Operations, Object-Based Manipulation, Conditional Logic, Loops, Methods. **Types, Objects, and Namespaces**: The Basics About Classes, Building a Basic Class, Value Types and Reference Types, Understanding Namespaces and Assemblies, Advanced Class Programming.	**12**
II	**Web Form Fundamentals**: Writing Code, Using the Code-Behind Class, Adding Event Handlers, Understanding the Anatomy of an ASP.NET Application, Introducing Server Controls, Using the Page Class, Using Application Events, Configuring an ASP.NET Application. **Form Controls**: Stepping Up to Web Controls, Web Control Classes, List Controls, Table Controls, Web Control Events and AutoPostBack, Validation, Understanding Validation, Using the Validation Controls, Rich Controls, The Calendar, The AdRotator, Pages with Multiple Views, User Controls and Graphics, User Controls, Dynamic Graphics, The Chart Control, Website Navigation: Site Maps, URL Mapping and Routing, The SiteMapPath Control, The TreeView Control, The Menu Control.	**12**
III	**Error Handling, Logging, and Tracing** : Avoiding Common Errors, Understanding Exception Handling, Handling Exceptions, Throwing Your Own Exceptions, Using Page Tracing **State Management** : Understanding the Problem of State, Using View State, Transferring Information Between Pages, Using Cookies, Managing Session State, Configuring Session State, Using Application State, Comparing State Management Options **Styles, Themes, and Master Pages** : Styles, Themes, Master Page Basics, Advanced Master Pages,	**12**
IV	**ADO.NET Fundamentals**: Understanding Databases, Configuring Your Database, Understanding SQL Basics, Understanding the Data Provider Model, Using Direct Data Access, Using Disconnected Data Access. **Data Binding**: Introducing Data Binding, Using Single-Value Data Binding, Using Repeated-Value Data Binding, Working with Data Source Controls,	**12**

	The Data Controls: The GridView, Formatting the GridView, Selecting a GridView Row, Editing with the GridView, Sorting and Paging the GridView, Using GridView Templates, The DetailsView and FormView	
V	XML: XML Explained, The XML Classes, XML Validation, XML Display and Transforms. Security Fundamentals: Understanding Security Requirements, Authentication and Authorization, Forms Authentication, Windows Authentication. ASP.NET AJAX: Understanding Ajax, Using Partial Refreshes, Using Progress Notification, Implementing Timed Refreshes, Working with the ASP.NET AJAX Control Toolkit.	12

Books and References:					
Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Beginning ASP.NET 4.5 in C#	Matthew MacDonald	Apress		2012
2.	C# 2015	Anne Bohem and Joel Murach	Murach	Third	2016
3.	Murach's ASP.NET 4.6 Web Programming in C#2015	Mary Delamater and Anne Bohem	SPD	Sixth	2016
4.	ASP.NET 4.0 programming	J. Kanjilal	Tata McGraw-Hill		2011
5.	Programming ASP.NET	D.Esposito	Microsoft Press (Dreamtech)		2011
6.	Beginning Visual C# 2010	K. Watson, C. Nagel, J.H Padderson, J.D. Reid, M.Skinner	Wrox (Wiley)		2010

B. Sc. (Information Technology)		Semester – V	
Course Name: Artificial Intelligence		**Course Code: USIT504 (Elective I)**	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	**Marks**
Evaluation System	**Theory Examination**	2½	75
	Internal	--	25

Unit	Details	Lectures
I	**Introduction:** What is Artificial Intelligence? Foundations of AI, history, the state of art AI today. **Intelligent Agents:** agents and environment, good behavior, nature of environment, the structure of agents.	12
II	**Solving Problems by Searching:** Problem solving agents, examples problems, searching for solutions, uninformed search, informed search strategies, heuristic functions. **Beyond Classical Search:** local search algorithms, searching with non-deterministic action, searching with partial observations, online search agents and unknown environments.	12
III	**Adversarial Search:** Games, optimal decisions in games, alpha-beta pruning, stochastic games, partially observable games, state-of-the-are game programs. **Logical Agents:** Knowledge base agents, The Wumpus world, logic, propositional logic, propositional theorem proving, effective propositional model checking, agents based on propositional logic.	12
IV	**First Order Logic:** Syntax and semantics, using First Order Logic, Knowledge engineering in First Order Logic. **Inference in First Order Logic:** propositional vs. First Order, unification and lifting, forward and backward chaining, resolution.	12
V	**Planning:** Definition of Classical Planning, Algorithms for planning as state space search, planning graphs, other classical planning approaches, analysis of planning approaches, Time, Schedules and resources, hierarchical planning, Planning and Acting in Nondeterministic Domains, multiagent planning, **Knowledge Representation:** Categories and Objects, events, mental events and objects, reasoning systems for categories, reasoning with default information, Internet shopping world	12

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Artificial Intelligence: A Modern Approach	Stuart Russel and Peter Norvig	Pearson	3rd	2015

2.	A First Course in Artificial Intelligence	Deepak Khemani	TMH	First	2017
3.	Artificial Intelligence: A Rational Approach	Rahul Deva	Shroff publishers	1st	2018
4.	Artificial Intelligence	Elaine Rich, Kevin Knight and Shivashankar Nair	TMH	3rd	2009
5.	Artificial Intelligence & Soft Computing for Beginners	Anandita Das Bhattacharjee	SPD	1st	2013

B. Sc. (Information Technology)		Semester – V	
Course Name: Linux System Administration		**Course Code: USIT505 (Elective I)**	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	**Marks**
Evaluation System	**Theory Examination**	2½	75
	Internal	--	25

Unit	Details	Lectures
I	**Introduction to Red Hat Enterprise Linux:** Linux, Open Source and Red Hat, Origins of Linux, Distributions, Duties of Linux System Administrator. **Command Line:** Working with the Bash Shell, Getting the Best of Bash, Useful Bash Key Sequences, Working with Bash History, Performing Basic File System Management Tasks, Working with Directories, Piping and Redirection, Finding Files **System Administration Tasks:** Performing Job Management Tasks, System and Process Monitoring and Management, Managing Processes with ps, Sending Signals to Processes with the kill Command, Using top to Show Current System Activity, Managing Process Niceness, Scheduling Jobs, Mounting Devices, Working with Links, Creating Backups, Managing Printers, Setting Up System Logging, Setting Up Rsyslog, Common Log Files, Setting Up Logrotate **Managing Software:** Understanding RPM, Understanding Meta Package Handlers, Creating Your Own Repositories, Managing Repositories, Installing Software with Yum, Querying Software, Extracting Files from RPM Packages	12
II	**Configuring and Managing Storage:** Understanding Partitions and Logical Volumes, Creating Partitions, Creating File Systems, File Systems Overview, Creating File Systems, Changing File System Properties, Checking the File System Integrity, Mounting File Systems Automatically Through fstab, Working with Logical Volumes, Creating Logical Volumes, Resizing Logical Volumes, Working with Snapshots, Replacing Failing Storage Devices, Creating Swap Space, Working with Encrypted Volumes **Connecting to the Network:** Understanding NetworkManager, Working with Services and Runlevels, Configuring the Network with NetworkManager, Working with system-config-network, NetworkManager Configuration Files, Network Service Scripts, Networking from the Command Line, Troubleshooting Networking, Setting Up IPv6, Configuring SSH, Enabling the SSH Server, Using the SSH Client, Using PuTTY on Windows Machines, Configuring Key-Based SSH Authentication, Using Graphical Applications with SSH, Using SSH Port Forwarding, Configuring VNC Server Access	12

	Working with Users, Groups, and Permissions: Managing Users and Groups, Commands for User Management, Managing Passwords, Modifying and Deleting User Accounts, Configuration Files, Creating Groups, Using Graphical Tools for User, and Group Management, Using External Authentication Sources, the Authentication Process, sssd, nsswitch, Pluggable Authentication Modules, Managing Permissions, the Role of Ownership, Basic Permissions: Read, Write, and Execute, Advanced Permissions, Working with Access Control Lists, Setting Default Permissions with umask, Working with Attributes	
III	**Securing Server with iptables:** Understanding Firewalls, Setting Up a Firewall with system-config-firewall, Allowing Services, Trusted Interfaces, Masquerading, Configuration Files, Setting Up a Firewall with iptables, Tables, Chains, and Rules, Composition of Rule, Configuration Example, Advanced iptables Configuration, Configuring Logging, The Limit Module, Configuring NAT **Setting Up Cryptographic Services:** Introducing SSL, Proof of Authenticity: the Certificate Authority, Managing Certificates with openssl, Creating a Signing Request, Working with GNU Privacy Guard, Creating GPG Keys, Key Transfer, Managing GPG Keys, Encrypting Files with GPG, GPG Signing, Signing RPM Files **Configuring Server for File Sharing:** What is NFS? Advantages and Disadvantages of NFS, Configuring NFS4, Setting Up NFSv4, Mounting an NFS Share, Making NFS Mounts Persistent, Configuring Automount, Configuring Samba, Setting Up a Samba File Server, Samba Advanced Authentication Options, Accessing Samba Shares, Offering FTP Services.	**12**
IV	**Configuring DNS and DHCP:** Introduction to DNS, The DNS Hierarchy, DNS Server Types, The DNS Lookup Process, DNS Zone Types, Setting Up a DNS Server, Setting Up a Cache-Only Name Server, Setting Up a Primary Name Server, Setting Up a Secondary Name Server, Understanding DHCP, Setting Up a DHCP Server **Setting Up a Mail Server:** Using the Message Transfer Agent, the Mail Delivery Agent, the Mail User Agent, Setting Up Postfix as an SMTP Server, Working with Mutt, Basic Configuration, Internet Configuration, Configuring Dovecot for POP and IMAP **Configuring Apache on Red Hat Enterprise Linux:** Configuring the Apache Web Server, Creating a Basic Website, Understanding the Apache Configuration Files, Apache Log Files, Working with Virtual Hosts, Securing the Web Server with TLS Certificates, Configuring Authentication, Setting Up Authentication with .htpasswd, Configuring LDAP Authentication, Setting Up MySQL	**12**

V	**Introducing Bash Shell Scripting:** Introduction, Elements of a Good Shell Script, Executing the Script, Working with Variables and Input, Understanding Variables, Variables, Subshells, and Sourcing, Working with Script Arguments, Asking for Input, Using Command Substitution, Substitution Operators, Changing Variable Content with Pattern Matching, Performing Calculations, Using Control Structures, Using if...then...else, Using case, Using while, Using until, Using for, Configuring booting with GRUB. **High-Availability Clustering:** High-Availability Clustering, The Workings of High Availability, High-Availability Requirements, Red Hat High-Availability Add-on Software, Components, Configuring Cluster-Based Services, Setting Up Bonding, Setting Up Shared Storage, Installing the Red Hat High Availability Add-On, Building the Initial State of the Cluster, Configuring Additional Cluster Properties, Configuring a Quorum Disk, Setting Up Fencing, Creating Resources and Services, Troubleshooting a Nonoperational Cluster, Configuring GFS2 File Systems **Setting Up an Installation Server:** Configuring a Network Server as an Installation Server, Setting Up a TFTP and DHCP Server for PXE Boot, Installing the TFTP Server, Configuring DHCP for PXE Boot, Creating the TFTP PXE Server Content, Creating a Kickstart File, Using a Kickstart File to Perform an Automated, Installation, Modifying the Kickstart File with, system-config-kickstart, Making Manual Modifications to the Kickstart File	12

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Red Hat Enterprise Linux 6 Administration	Sander van Vugt	John Wiley and Sons		2013
2.	Red hat Linux Networking and System Administration	Terry Collings and Kurt Wall	Wiley	3rd	
3.	Linux Administration: A Beginner's Guide	Wale Soyinka	TMH	Fifth Edition	

B. Sc. (Information Technology)		Semester – V	
Course Name: Enterprise Java		**Course Code: USIT506 (Elective II)**	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	**Marks**
Evaluation System	**Theory Examination**	2½	75
	Internal	--	25

Unit	Details	Lectures
I	**Understanding Java EE:** What is an Enterprise Application? What is java enterprise edition? Java EE Technologies, Java EE evolution, Glassfish server **Java EE Architecture, Server and Containers:** Types of System Architecture, Java EE Server, Java EE Containers. **Introduction to Java Servlets:** The Need for Dynamic Content, Java Servlet Technology, Why Servlets? What can Servlets do? **Servlet API and Lifecycle:** Java Servlet API, The Servlet Skeleton, The Servlet Life Cycle, A Simple Welcome Servlet **Working With Servlets:** Getting Started, Using Annotations Instead of Deployment Descriptor. **Working with Databases:** What Is JDBC? JDBC Architecture, Accessing Database, The Servlet GUI and Database Example.	**12**
II	**Request Dispatcher:** Resquestdispatcher Interface, Methods of Requestdispatcher, Requestdispatcher Application. **COOKIES:** Kinds Of Cookies, Where Cookies Are Used? Creating Cookies Using Servlet, Dynamically Changing The Colors Of A Page **SESSION:** What Are Sessions? Lifecycle Of Http Session, Session Tracking With Servlet API, A Servlet Session Example **Working With Files:** Uploading Files, Creating an Upload File Application, Downloading Files, Creating a Download File Application. **Working With Non-Blocking I/O:** Creating a Non-Blocking Read Application, Creating The Web Application, Creating Java Class, Creating Servlets, Retrieving The File, Creating index.jsp	**12**
III	**Introduction To Java Server Pages:** Why use Java Server Pages? Disadvantages Of JSP, JSP v\s Servlets, Life Cycle of a JSP Page, How does a JSP function? How does JSP execute? About Java Server Pages **Getting Started With Java Server Pages:** Comments, JSP Document, JSP Elements, JSP GUI Example. **Action Elements:** Including other Files, Forwarding JSP Page to Another Page, Passing Parameters for other Actions, Loading a Javabean. **Implicit Objects, Scope And El Expressions:** Implicit Objects, Character Quoting Conventions, Unified Expression Language [Unified El], Expression Language.	**12**

	Java Server Pages Standard Tag Libraries: What is wrong in using JSP Scriptlet Tags? How JSTL Fixes JSP Scriptlet's Shortcomings? Disadvantages Of JSTL, Tag Libraries.	
IV	**Introduction To Enterprise Javabeans:** Enterprise Bean Architecture, Benefits of Enterprise Bean, Types of Enterprise Bean, Accessing Enterprise Beans, Enterprise Bean Application, Packaging Enterprise Beans **Working With Session Beans:** When to use Session Beans? Types of Session Beans, Remote and Local Interfaces, Accessing Interfaces, Lifecycle of Enterprise Beans, Packaging Enterprise Beans, Example of Stateful Session Bean, Example of Stateless Session Bean, Example of Singleton Session Beans. **Working with Message Driven Beans:** Lifecycle of a Message Driven Bean, Uses of Message Driven Beans, The Message Driven Beans Example. **Interceptors:** Request And Interceptor, Defining An Interceptor, AroundInvoke Method, Applying Interceptor, Adding An Interceptor To An Enterprise Bean, Build and Run the Web Application. **Java Naming and Directory Interface**: What is Naming Service? What is Directory Service? What is Java Naming and Directory interface? Basic Lookup, JNDI Namespace in Java EE, Resources and JNDI, Datasource Resource Definition in Java EE.	**12**
V	**Persistence, Object/Relational Mapping And JPA:** What is Persistence? Persistence in Java, Current Persistence Standards in Java, Why another Persistence Standards? Object/Relational Mapping, **Introduction to Java Persistence API:** The Java Persistence API, JPA, ORM, Database and the Application, Architecture of JPA, How JPA Works? JPA Specifications. **Writing JPA Application:** Application Requirement Specifications, Software Requirements, The Application Development Approach, Creating Database And Tables in Mysql, Creating a Web Application, Adding the Required Library Files, Creating a Javabean Class, Creating Persistence Unit [Persistence.Xml], Creating JSPS, The JPA Application Structure, Running The JPA Application. **Introduction to Hibernate:** What is Hibernate? Why Hibernate? Hibernate, Database and The Application, Components of Hibernate, Architecture of Hibernate, How Hibernate Works? **Writing Hibernate Application:** Application Requirement Specifications, Software Requirements, The Application Development Approach, Creating Database and Tables in Mysql, Creating a Web Application, Adding The Required Library Files, Creating a Javabean Class, Creating Hibernate Configuration File, Adding a Mapping Class, Creating JSPS, Running The Hibernate Application.	**12**

Books and References:					
Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Java EE 7 For Beginners	Sharanam Shah, Vaishali Shah	SPD	First	2017
2.	Java EE 8 Cookbook: Build reliable applications with the most robust and mature technology for enterprise development	Elder Moraes	Packt	First	2018
3.	Advanced Java Programming	Uttam Kumar Roy	Oxford Press		2015

B. Sc. (Information Technology)		Semester – V	
Course Name: Next Generation Technologies		Course Code: USIT507 (Elective II)	
Periods per week (1 Period is 50 minutes),		5	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2½	75
	Internal	--	25

Unit	Details	Lectures
I	**Big Data:** Getting Started, Big Data, Facts About Big Data, Big Data Sources, Three Vs of Big Data, Volume, Variety, Velocity, Usage of Big Data, Visibility, Discover and Analyze Information, Segmentation and Customizations, Aiding Decision Making, Innovation, Big Data Challenges, Policies and Procedures, Access to Data, Technology and Techniques, Legacy Systems and Big Data, Structure of Big Data, Data Storage, Data Processing, Big Data Technologies **NoSQL:** SQL, NoSQL, Definition, A Brief History of NoSQL, ACID vs. BASE, CAP Theorem (Brewer's Theorem), The BASE, NoSQL Advantages and Disadvantages, Advantages of NoSQL, Disadvantages of NoSQL, SQL vs. NoSQL Databases, Categories of NoSQL Databases **Introducing MongoDB:** History, MongoDB Design Philosophy, Speed, Scalability, and Agility, Non-Relational Approach, JSON-Based Document Store, Performance vs. Features, Running the Database Anywhere, SQL Comparison	**12**
II	**The MongoDB Data Model:** The Data Model, JSON and BSON, The Identifier (_id), Capped Collection, Polymorphic Schemas, Object-Oriented Programming, Schema Evolution **Using MongoDB Shell:** Basic Querying, Create and Insert, Explicitly Creating Collections, Inserting Documents Using Loop, Inserting by Explicitly Specifying _id, Update, Delete, Read, Using Indexes, Stepping Beyond the Basics, Using Conditional Operators, Regular Expressions, MapReduce, aggregate(), Designing an Application's Data Model, Relational Data Modeling and Normalization, MongoDB Document Data Model Approach **MongoDB Architecture:** Core Processes, mongod, mongo, mongos, MongoDB Tools, Standalone Deployment, Replication, Master/Slave Replication, Replica Set, Implementing Advanced Clustering with Replica Sets, Sharding, Sharding Components, Data Distribution Process, Data Balancing Process, Operations, Implementing Sharding, Controlling Collection Distribution (Tag-Based Sharding), Points to	**12**

	Remember When Importing Data in a ShardedEnvironment, Monitoring for Sharding, Monitoring the Config Servers, Production Cluster Architecture, Scenario 1, Scenario 2, Scenario 3, Scenario 4	
III	**MongoDB Storage Engine:** Data Storage Engine, Data File (Relevant for MMAPv1), Namespace (.ns File), Data File (Relevant for WiredTiger), Reads and Writes, How Data Is Written Using Journaling, GridFS – The MongoDB File System, The Rationale of GridFS, GridFSunder the Hood, Using GridFS, Indexing, Types of Indexes, Behaviors and Limitations **MongoDB Use Cases:** Use Case 1 -Performance Monitoring, Schema Design, Operations, Sharding, Managing the Data, Use Case 2 – Social Networking, Schema Design, Operations, Sharding **MongoDB Limitations:** MongoDB Space Is Too Large (Applicable for MMAPv1), Memory Issues (Applicable for Storage Engine MMAPv1), 32-bit vs. 64-bit, BSON Documents, Namespaces Limits, Indexes Limit, Capped Collections Limit - Maximum Number of Documents in a Capped Collection, Sharding Limitations, Shard Early to Avoid Any Issues, Shard Key Can't Be Updated, Shard Collection Limit, Select the Correct Shard Key, Security Limitations, No Authentication by Default, Traffi c to and from MongoDB Isn't Encrypted, Write and Read Limitations, Case-Sensitive Queries, Type-Sensitive Fields, No JOIN, Transactions, MongoDB Not Applicable Range **MongoDB Best Practices:** Deployment, Hardware Suggestions from the MongoDB Site, Few Points to be Noted, Coding, Application Response Time Optimization, Data Safety, Administration, Replication Lag, Sharding, Monitoring	**12**
IV	**The End of Disk? SSD and In-Memory Databases:** The End of Disk?, Solid State Disk, The Economics of Disk, SSD-Enabled Databases, In-Memory Databases, TimesTen, Redis, SAP HANA, VoltDB, Oracle 12c "in-Memory Database, Berkeley Analytics Data Stack and Spark, Spark Architecture **jQuery:** Introduction, Traversing the DOM, DOM Manipulation with jQuery, Events, Ajax with jQuery, jQuery Plug-ins, jQuery Image Slider	**12**
V	**JSON:** Introduction, JSON Grammar, JSON Values, JSON Tokens, Syntax, JSON vs XML, Data Types, Objects, Arrays, Creating JSON, JSON Object, Parsing JSON, Persisting JSON, Data Interchange, JSON PHP, JSON HTML, JSONP	**12**

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Practical MongoDB	Shakuntala Gupta Edward Navin Sabharwal	Apress		
2.	Beginning jQuery	Jack Franklin Russ Ferguson	Apress	Second	
3.	Next Generation Databases	Guy Harrison	Apress		
4.	Beginning JSON	Ben Smith	Apress		

B. Sc. (Information Technology)		Semester – V	
Course Name: Project Dissertation		**Course Code: USIT5P1**	
Periods per week (1 Period is 50 minutes)		**3**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Practical Examination**	**2½**	**50**
	Internal	**--**	**--**

The details are given in Appendix – I

B. Sc. (Information Technology)		Semester – V	
Course Name: Internet of Things Practical		**Course Code: USIT5P2**	
Periods per week (1 Period is 50 minutes)		**3**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Practical Examination**	**2½**	**50**
	Internal	**--**	**--**

Practical No	Details
0	Starting Raspbian OS, Familiarising with Raspberry Pi Components and interface, Connecting to ethernet, Monitor, USB.
1	Displaying different LED patterns with Raspberry Pi.
2	Displaying Time over 4-Digit 7-Segment Display using Raspberry Pi
3	Raspberry Pi Based Oscilloscope
4	Controlling Raspberry Pi with WhatsApp.
5	Setting up Wireless Access Point using Raspberry Pi
6	Fingerprint Sensor interfacing with Raspberry Pi
7	Raspberry Pi GPS Module Interfacing
8	IoT based Web Controlled Home Automation using Raspberry Pi
9	Visitor Monitoring with Raspberry Pi and Pi Camera
10	Interfacing Raspberry Pi with RFID.
11	Building Google Assistant with Raspberry Pi.
12	Installing Windows 10 IoT Core on Raspberry Pi

Raspberry Pi Kits and components should be made available in the ratio of 1 kit : 3 students minimum.

B. Sc. (Information Technology)		Semester – V	
Course Name: Advanced Web Programming Practical		**Course Code: USIT5P3**	
Periods per week (1 Period is 50 minutes)		**3**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Practical Examination**	**2½**	**50**
	Internal	**--**	**--**

List of Practical	
1.	**Working with basic C# and ASP .NET**
a.	Create an application that obtains four int values from the user and displays the product.
b.	Create an application to demonstrate string operations.
c.	Create an application that receives the (Student Id, Student Name, Course Name, Date of Birth) information from a set of students. The application should also display the information of all the students once the data entered.
	Create an application to demonstrate following operations i. Generate Fibonacci series. ii. Test for prime numbers. iii. Test for vowels. iv. Use of foreach loop with arrays v. Reverse a number and find sum of digits of a number.
2.	**Working with Object Oriented C# and ASP .NET**
a.	Create simple application to perform following operations i. Finding factorial Value ii. Money Conversion iii. Quadratic Equation iv. Temperature Conversion
b.	Create simple application to demonstrate use of following concepts i. Function Overloading ii. Inheritance (all types) iii. Constructor overloading iv. Interfaces
c.	Create simple application to demonstrate use of following concepts i. Using Delegates and events ii. Exception handling
3.	**Working with Web Forms and Controls**
a.	Create a simple web page with various sever controls to demonstrate setting and use of their properties. (Example : AutoPostBack)
b.	Demonstrate the use of Calendar control to perform following operations. a) Display messages in a calendar control b) Display vacation in a calendar control c) Selected day in a calendar control using style d) Difference between two calendar dates
c.	Demonstrate the use of Treeview control perform following operations. a) Treeview control and datalist b) Treeview operations
4.	**Working with Form Controls**
a.	Create a Registration form to demonstrate use of various Validation controls.
b.	Create Web Form to demonstrate use of Adrotator Control.
c.	Create Web Form to demonstrate use User Controls.

5.	**Working with Navigation, Beautification and Master page.**
a.	Create Web Form to demonstrate use of Website Navigation controls and Site Map.
b.	Create a web application to demonstrate use of Master Page with applying Styles and Themes for page beautification.
c.	Create a web application to demonstrate various states of ASP.NET Pages.
6.	**Working with Database**
a.	Create a web application bind data in a multiline textbox by querying in another textbox.
b.	Create a web application to display records by using database.
c.	Demonstrate the use of Datalist link control.
7.	**Working with Database**
a.	Create a web application to display Databinding using dropdownlist control.
b.	Create a web application for to display the phone no of an author using database.
c.	Create a web application for inserting and deleting record from a database. (Using Execute-Non Query).
8.	**Working with data controls**
a.	Create a web application to demonstrate various uses and properties of SqlDataSource.
b.	Create a web application to demonstrate data binding using DetailsView and FormView Control.
c.	Create a web application to display Using Disconnected Data Access and Databinding using GridView.
9.	**Working with GridView control**
a.	Create a web application to demonstrate use of GridView control template and GridView hyperlink.
b.	Create a web application to demonstrate use of GridView button column and GridView events.
c.	Create a web application to demonstrate GridView paging and Creating own table format using GridView.
10.	**Working with AJAX and XML**
a.	Create a web application to demonstrate reading and writing operation with XML.
b.	Create a web application to demonstrate Form Security and Windows Security with proper Authentication and Authorization properties.
c.	Create a web application to demonstrate use of various Ajax controls.
11.	**Programs to create and use DLL**

B. Sc. (Information Technology)		Semester – V	
Course Name: Artificial Intelligence Practical		Course Code: USIT5P4 (Elective I)	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2½	50
	Internal	--	--

Practical No		Details
1	a	Write a program to implement depth first search algorithm.
	b	Write a program to implement breadth first search algorithm.
2	a	Write a program to simulate 4-Queen / N-Queen problem.
	b	Write a program to solve tower of Hanoi problem.
3	a	Write a program to implement alpha beta search.
	b	Write a program for Hill climbing problem.
4	a	Write a program to implement A* algorithm.
	b	Write a program to implement AO* algorithm.
5	a	Write a program to solve water jug problem.
	b	Design the simulation of tic – tac – toe game using min-max algorithm.
6	a	Write a program to solve Missionaries and Cannibals problem.
	b	Design an application to simulate number puzzle problem.
7	a	Write a program to shuffle Deck of cards.
	b	Solve traveling salesman problem using artificial intelligence technique.
8	a	Solve the block of World problem.
	b	Solve constraint satisfaction problem
9	a	Derive the expressions based on Associative law
	b	Derive the expressions based on Distributive law
10	a	Write a program to derive the predicate. (for e.g.: Sachin is batsman , batsman is cricketer) - > Sachin is Cricketer.
	b	Write a program which contains three predicates: male, female, parent. Make rules for following family relations: father, mother, grandfather,grandmother, brother, sister, uncle, aunt, nephew and niece, cousin. Question: i. Draw Family Tree. ii. Define: Clauses, Facts, Predicates and Rules with conjunction and disjunction

The practicals can be implemented in C / C++ / Java/ Python / R /Prolog / LISP or any other language.

B. Sc. (Information Technology)		Semester – V	
Course Name: Linux System Administration		Course Code: USIT5P5 (Elective I)	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2½	50
	Internal	--	--

Practical No	Details
0	**Installation of RHEL 6.X**
1	**Graphical User Interface and Command Line Interface and Processes**
a	Exploring the Graphical Desktop
b	The Command Line Interface
c	Managing Processes
2	**Storage Devices and Links, Backup and Repository**
b	Working with Storage Devices and Links
a	Making a Backup
b	Creating a Repository
3	**Working with RPMsm Storage and Networking**
a	Using Query Options
b	Extracting Files From RPMs
c	Configuring and Managing Storage
d	Connecting to the Network
4	**Working with Users, Groups, and Permissions**
5	**Firewall and Cryptographic services**
a	Securing Server with iptables
b	Setting Up Cryptographic Services
6	**Configuring Server for File Sharing**
a	Configuring NFS Server and Client
b	Configuring Samba
c	Configuring FTP
7	**DNS, DHCP and Mail Server**
a	Configuring DNS
b	Configuring DHCP
c	Setting Up a Mail Server

8	**Web Server**
a	Configuring Apache on Red Hat Enterprise Linux
b	Writing a Script to Monitor Activity on the Apache Web Server
c	Using the select Command
9	**Shell Scripts and High-Availability Clustering**
a	Writing Shell Scripts
b	Configuring Booting with GRUB
c	Configuring High Availability Clustering
10	**Setting Up an Installation Server**
a	Configuring Network Server as an Installation Server
b	Setting Up a TFTP and DHCP Server for PXE Boot

B. Sc. (Information Technology)		Semester – V	
Course Name: Enterprise Java		**Course Code: USIT5P6 (Elective II)**	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	**Marks**
Evaluation System	**Practical Examination**	2½	50
	Internal	--	--

List of Practical	
1.	**Implement the following Simple Servlet applications.**
a.	Create a simple calculator application using servlet.
b.	Create a servlet for a login page. If the username and password are correct then it says message "Hello <username>" else a message "login failed"
c.	Create a registration servlet in Java using JDBC. Accept the details such as Username, Password, Email, and Country from the user using HTML Form and store the registration details in the database.
2.	**Implement the following Servlet applications with Cookies and Sessions.**
a.	Using Request Dispatcher Interface create a Servlet which will validate the password entered by the user, if the user has entered "Servlet" as password, then he will be forwarded to Welcome Servlet else the user will stay on the index.html page and an error message will be displayed.
b.	Create a servlet that uses Cookies to store the number of times a user has visited servlet.
c.	Create a servlet demonstrating the use of session creation and destruction. Also check whether the user has visited this page first time or has visited earlier also using sessions.
3.	**Implement the Servlet IO and File applications.**
a.	Create a Servlet application to upload and download a file.
b.	Develop Simple Servlet Question Answer Application using Database.
c.	Create simple Servlet application to demonstrate Non-Blocking Read Operation.
4.	**Implement the following JSP applications.**
a.	Develop a simple JSP application to display values obtained from the use of intrinsic objects of various types.
b.	Develop a simple JSP application to pass values from one page to another with validations. (Name-txt, age-txt, hobbies-checkbox, email-txt, gender-radio button).
c.	Create a registration and login JSP application to register and authenticate the user based on username and password using JDBC.
5.	**Implement the following JSP JSTL and EL Applications.**
a.	Create an html page with fields, eno, name, age, desg, salary. Now on submit this data to a JSP page which will update the employee table of database with matching eno.
b.	Create a JSP page to demonstrate the use of Expression language.
c.	Create a JSP application to demonstrate the use of JSTL.

6.	**Implement the following EJB Applications.**
a.	Create a Currency Converter application using EJB.
b.	Develop a Simple Room Reservation System Application Using EJB.
c.	Develop simple shopping cart application using EJB [Stateful Session Bean].
7.	**Implement the following EJB applications with different types of Beans.**
a.	Develop simple EJB application to demonstrate Servlet Hit count using Singleton Session Beans.
b.	Develop simple visitor Statistics application using Message Driven Bean [Stateless Session Bean].
c.	Develop simple Marks Entry Application to demonstrate accessing Database using EJB.
8.	**Implement the following JPA applications.**
a.	Develop a simple Inventory Application Using JPA.
b.	Develop a Guestbook Application Using JPA.
c.	Create simple JPA application to store and retrieve Book details.
9.	**Implement the following JPA applications with ORM and Hibernate.**
a.	Develop a JPA Application to demonstrate use of ORM associations.
b.	Develop a Hibernate application to store Feedback of Website Visitor in MySQL Database.
c.	Develop a Hibernate application to store and retrieve employee details in MySQL Database.
10.	**Implement the following Hibernate applications.**
a.	Develop an application to demonstrate Hibernate One- To -One Mapping Using Annotation.
b.	Develop Hibernate application to enter and retrieve course details with ORM Mapping.
c.	Develop a five page web application site using any two or three Java EE Technologies.

B. Sc. (Information Technology)		Semester – V	
Course Name: Next Generation Technologies Practical		**Course Code: USIT5P7 (Elective II)**	
Periods per week (1 Period is 50 minutes)		**3**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Practical Examination**	2½	50
	Internal	--	--

Practical No	Details
1	**MongoDB Basics**
a	Write a MongoDB query to create and drop database.
b	Write a MongoDB query to create, display and drop collection
c	Write a MongoDB query to insert, query, update and delete a document.
2	**Simple Queries with MongoDB**
3	**Implementing Aggregation**
a	Write a MongoDB query to use sum, avg, min and max expression.
b	Write a MongoDB query to use push and addToSet expression.
c	Write a MongoDB query to use first and last expression.
4	**Replication, Backup and Restore**
a	Write a MongoDB query to create Replica of existing database.
b	Write a MongoDB query to create a backup of existing database.
c	Write a MongoDB query to restore database from the backup.
5	**Java and MongoDB**
a	Connecting Java with MongoDB and inserting, retrieving, updating and deleting.
6	**PHP and MongoDB**
a	Connecting PHP with MongoDB and inserting, retrieving, updating and deleting.
7	**Python and MongoDB**
a	Connecting Python with MongoDB and inserting, retrieving, updating and deleting.
8	**Programs on Basic jQuery**
a	jQuery Basic, jQuery Events
b	jQuery Selectors, jQuery Hide and Show effects
c	jQuery fading effects, jQuery Sliding effects

9	**jQuery Advanced**
a	jQuery Animation effects, jQuery Chaining
b	jQuery Callback, jQuery Get and Set Contents
c	jQuery Insert Content, jQuery Remove Elements and Attribute
10	**JSON**
a	Creating JSON
b	Parsing JSON
c	Persisting JSON
11	**Create a JSON file and import it to MongoDB**
a	Export MongoDB to JSON.
b	Write a MongoDB query to delete JSON object from MongoDB

SEMESTER VI

B. Sc. (Information Technology)		Semester – VI	
Course Name: Software Quality Assurance		**Course Code: USIT601**	
Periods per week (1 Period is 50 minutes)		**5**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	**2½**	**75**
	Internal	**--**	**25**

Unit	Details	Lectures
I	**Introduction to Quality:** Historical Perspective of Quality, What is Quality? (Is it a fact or perception?), Definitions of Quality, Core Components of Quality, Quality View, Financial Aspect of Quality, Customers, Suppliers and Processes, Total Quality Management (TQM), Quality Principles of Total Quality Management, Quality Management Through Statistical Process Control, Quality Management Through Cultural Changes, Continual (Continuous) Improvement Cycle, Quality in Different Areas, Benchmarking and Metrics, Problem Solving Techniques, Problem Solving Software Tools. **Software Quality:** Introduction, Constraints of Software Product Quality Assessment, Customer is a King, Quality and Productivity Relationship, Requirements of a Product, Organisation Culture, Characteristics of Software, Software Development Process, Types of Products, Schemes of Criticality Definitions, Problematic Areas of Software Development Life Cycle, Software Quality Management, Why Software Has Defects? Processes Related to Software Quality, Quality Management System Structure, Pillars of Quality Management System, Important Aspects of Quality Management.	**12**
II	**Fundamentals of testing:** Introduction, Necessity of testing, What is testing? Fundamental test process, The psychology of testing, Historical Perspective of Testing, Definitions of Testing, Approaches to Testing, Testing During Development Life Cycle, Requirement Traceability Matrix, Essentials of Software Testing, Workbench, Important Features of Testing Process, Misconceptions About Testing, Principles of Software Testing, Salient Features of Good Testing, Test Policy, Test Strategy or Test Approach, Test Planning, Testing Process and Number of Defects Found in Testing, Test Team Efficiency, Mutation Testing, Challenges in Testing, Test Team Approach, Process Problems Faced by Testing, Cost Aspect of Testing, Establishing Testing Policy, Methods, Structured Approach to Testing, Categories of Defect, Defect, Error, or Mistake in Software, Developing Test Strategy, Developing Testing Methodologies (Test Plan), Testing Process, Attitude Towards Testing (Common People Issues), Test Methodologies/Approaches, People Challenges in Software Testing, Raising Management Awareness for Testing, Skills Required by Tester,	**12**

	Testing throughout the software life cycle, Software development models, Test levels, Test types, the targets of testing, Maintenance testing	
III	**Unit Testing: Boundary Value Testing:** Normal Boundary Value Testing, Robust Boundary Value Testing, Worst-Case Boundary Value Testing, Special Value Testing, Examples, Random Testing, Guidelines for Boundary Value Testing, **Equivalence Class Testing:** Equivalence Classes, Traditional Equivalence Class Testing, Improved Equivalence Class Testing, Edge Testing, Guidelines and Observations. **Decision Table–Based Testing:** Decision Tables, Decision Table Techniques, Cause-and-Effect Graphing, Guidelines and Observations, **Path Testing:** Program Graphs, DD-Paths, Test Coverage Metrics, Basis Path Testing, Guidelines and Observations, **Data Flow Testing:** Define/Use Testing, Slice-Based Testing, Program Slicing Tools.	**12**
IV	**Software Verification and Validation:** Introduction, Verification, Verification Workbench, Methods of Verification, Types of reviews on the basis od Stage Phase, Entities involved in verification, Reviews in testing lifecycle, Coverage in Verification, Concerns of Verification, Validation, Validation Workbench, Levels of Validation, Coverage in Validation, Acceptance Testing, Management of Verification and Validation, Software development verification and validation activities. **V-test Model:** Introduction, V-model for software, Testing during Proposal stage, Testing during requirement stage, Testing during test planning phase, Testing during design phase, Testing during coding, VV Model, Critical Roles and Responsibilities.	**12**
V	**Levels of Testing:** Introduction, Proposal Testing, Requirement Testing, Design Testing, Code Review, Unit Testing, Module Testing, Integration Testing, Big-Bang Testing, Sandwich Testing, Critical Path First, Sub System Testing, System Testing, Testing Stages. **Special Tests:** Introduction, GUI testing, Compatibility Testing, Security Testing, Performance Testing, Volume Testing, Stress Testing, Recovery Testing, Installation Testing, Requirement Testing, Regression Testing, Error Handling Testing, Manual Support Testing, Intersystem Testing, Control Testing, Smoke Testing, Adhoc Testing, Parallel Testing, Execution Testing, Operations Testing, Compliance Testing, Usability Testing, Decision Table Testing, Documentation Testing, Training testing, Rapid Testing, Control flow graph, Generating tests on the basis of Combinatorial Designs, State Graph, Risk Associated with New Technologies, Process maturity level of Technology, Testing Adequacy of Control in New technology usage, Object Oriented Application Testing, Testing of Internal Controls, COTS Testing, Client Server Testing, Web Application Testing, Mobile Application Testing, eBusiness eCommerce Testing, Agile Development Testing, Data Warehousing Testing.	**12**

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Software Testing and Continuous Quality Improvement	William E. Lewis	CRC Press	Third	2016
2	Software Testing: Principles, Techniques and Tools	M. G. Limaye	TMH		2017
3.	Foundations of Software Testing	Dorothy Graham, Erik van Veenendaal, Isabel Evans, Rex Black	Cengage Learning	3rd	
4.	Software Testing: A Craftsman's Approach	Paul C. Jorgenson	CRC Press	4th	2017

B. Sc. (Information Technology)		Semester – VI	
Course Name: Security in Computing		**Course Code: USIT602**	
Periods per week (1 Period is 50 minutes)		**5**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	2½	75
	Internal	--	25

Unit	Details	Lectures
I	**Information Security Overview** : The Importance of Information Protection, The Evolution of Information Security, Justifying Security Investment, Security Methodology, How to Build a Security Program, The Impossible Job, The Weakest Link, Strategy and Tactics, Business Processes vs. Technical Controls. **Risk Analysis**: Threat Definition, Types of Attacks, Risk Analysis. **Secure Design Principles**: The CIA Triad and Other Models, Defense Models, Zones of Trust, Best Practices for Network Defense.	**12**
II	**Authentication and Authorization**: Authentication, Authorization **Encryption**: A Brief History of Encryption, Symmetric-Key Cryptography, Public Key Cryptography, Public Key Infrastructure. **Storage Security**: Storage Security Evolution, Modern Storage Security, Risk Remediation, Best Practices. **Database Security**: General Database Security Concepts, Understanding Database Security Layers, Understanding Database-Level Security, Using Application Security, Database Backup and Recovery, Keeping Your Servers Up to Date, Database Auditing and Monitoring.	**12**
III	**Secure Network Design**: Introduction to Secure Network Design, Performance, Availability, Security. **Network Device Security**: Switch and Router Basics, Network Hardening. **Firewalls**: Overview, The Evolution of Firewalls, Core Firewall Functions, Additional Firewall Capabilities, Firewall Design. **Wireless Network Security**: Radio Frequency Security Basics, Data-Link Layer Wireless Security Features, Flaws, and Threats, Wireless Vulnerabilities and Mitigations, Wireless Network Hardening Practices and Recommendations, Wireless Intrusion Detection and Prevention, Wireless Network Positioning and Secure Gateways.	**12**
IV	**Intrusion Detection and Prevention Systems**: IDS Concepts, IDS Types and Detection Models, IDS Features, IDS Deployment Considerations, Security Information and Event Management (SIEM). **Voice over IP (VoIP) and PBX Security**: Background, VoIP Components, VoIP Vulnerabilities and Countermeasures, PBX, TEM: Telecom Expense Management. **Operating System Security Models**: Operating System Models, Classic Security Models, Reference Monitor, Trustworthy Computing, International Standards for Operating System Security.	**12**

| V | **Virtual Machines and Cloud Computing**: Virtual Machines, Cloud Computing. **Secure Application Design**: Secure Development Lifecycle, Application Security Practices, Web Application Security, Client Application Security, Remote Administration Security. **Physical Security**: Classification of Assets, Physical Vulnerability Assessment, Choosing Site Location for Security, Securing Assets: Locks and Entry Controls, Physical Intrusion Detection. | **12** |

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	The Complete Reference: Information Security	Mark Rhodes-Ousley	McGraw-Hill	2nd	2013
2.	Essential Cybersecurity Science	Josiah Dykstra	O'Reilly	Fifth	2017
3.	Principles of Computer Security: CompTIA Security+ and Beyond	Wm.Arthur Conklin, Greg White	McGraw Hill	Second	2010

B. Sc. (Information Technology)		Semester – VI	
Course Name: Business Intelligence		**Course Code: USIT603**	
Periods per week (1 Period is 50 minutes)		**5**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	2½	75
	Internal	--	25

Unit	Details	Lectures
I	**Business intelligence:** Effective and timely decisions, Data, information and knowledge, The role of mathematical models, Business intelligence architectures, Ethics and business intelligence **Decision support systems:** Definition of system, Representation of the decision-making process, Evolution of information systems, Definition of decision support system, Development of a decision support system	**12**
II	**Mathematical models for decision making:** Structure of mathematical models, Development of a model, Classes of models **Data mining:** Definition of data mining, Representation of input data , Data mining process, Analysis methodologies **Data preparation**: Data validation, Data transformation, Data reduction	**12**
III	**Classification**: Classification problems, Evaluation of classification models, Bayesian methods, Logistic regression, Neural networks, Support vector machines **Clustering:** Clustering methods, Partition methods, Hierarchical methods, Evaluation of clustering models	**12**
IV	**Business intelligence applications:** **Marketing models**: Relational marketing, Sales force management, **Logistic and production models**: Supply chain optimization, Optimization models for logistics planning, Revenue management systems. **Data envelopment analysis**: Efficiency measures, Efficient frontier, The CCR model, Identification of good operating practices	**12**
V	**Knowledge Management:** Introduction to Knowledge Management, Organizational Learning and Transformation, Knowledge Management Activities, Approaches to Knowledge Management, Information Technology (IT) In Knowledge Management, Knowledge Management Systems Implementation, Roles of People in Knowledge Management **Artificial Intelligence and Expert Systems:** Concepts and Definitions of Artificial Intelligence, Artificial Intelligence Versus Natural Intelligence, Basic Concepts of Expert Systems, Applications of Expert Systems, Structure of Expert Systems, Knowledge Engineering, Development of Expert Systems	**12**

Books and References:					
Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Business Intelligence: Data Mining and Optimization for Decision Making	Carlo Vercellis	Wiley	First	2009
2.	Decision support and Business Intelligence Systems	Efraim Turban, Ramesh Sharda, Dursun Delen	Pearson	Ninth	2011
3.	Fundamental of Business Intelligence	Grossmann W, Rinderle-Ma	Springer	First	2015

B. Sc. (Information Technology)		Semester – VI	
Course Name: Principles of Geographic Information Systems		Course Code: USIT604 (Elective I)	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2½	75
	Internal	--	25

Unit	Details	Lectures
I	**A Gentle Introduction to GIS** **The nature of GIS:** Some fundamental observations, Defining GIS, GISystems, GIScience and GIApplications, Spatial data and Geoinformation. **The real world and representations of it:** Models and modelling, Maps, Databases, Spatial databases and spatial analysis **Geographic Information and Spatial Database** **Models and Representations of the real world** **Geographic Phenomena:** Defining geographic phenomena, types of geographic phenomena, Geographic fields, Geographic objects, Boundaries **Computer Representations of Geographic Information:** Regular tessellations, irregular tessellations, Vector representations, Topology and Spatial relationships, Scale and Resolution, Representation of Geographic fields, Representation of Geographic objects **Organizing and Managing Spatial Data** **The Temporal Dimension**	12
II	**Data Management and Processing Systems** **Hardware and Software Trends** **Geographic Information Systems:** GIS Software, GIS Architecture and functionality, Spatial Data Infrastructure (SDI) **Stages of Spatial Data handling:** Spatial data handling and preparation, Spatial Data Storage and maintenance, Spatial Query and Analysis, Spatial Data Presentation. **Database management Systems:** Reasons for using a DBMS, Alternatives for data management, The relational data model, Querying the relational database. **GIS and Spatial Databases:** Linking GIS and DBMS, Spatial database functionality.	12
III	**Spatial Referencing and Positioning** **Spatial Referencing:** Reference surfaces for mapping, Coordinate Systems, Map Projections, Coordinate Transformations	12

	Satellite-based Positioning: Absolute positioning, Errors in absolute positioning, Relative positioning, Network positioning, code versus phase measurements, Positioning technology **Data Entry and Preparation** **Spatial Data Input:** Direct spatial data capture, Indirect spatial data captiure, Obtaining spatial data elsewhere **Data Quality:** Accuracy and Positioning, Positional accuracy, Attribute accuracy, Temporal accuracy, Lineage, Completeness, Logical consistency **Data Preparation:** Data checks and repairs, Combining data from multiple sources **Point Data Transformation:** Interpolating discrete data, Interpolating continuous data	
IV	**Spatial Data Analysis** **Classification of analytical GIS Capabilities** **Retrieval, classification and measurement:** Measurement, Spatial selection queries, Classification **Overlay functions:** Vector overlay operators,Raster overlay operators **Neighbourhood functions:** Proximity computations, Computation of diffusion, Flow computation, Raster based surface analysis **Analysis:** Network analysis, interpolation, terrain modeling **GIS and Application models:** GPS, Open GIS Standards, GIS Applications and Advances **Error Propagation in spatial data processing:** How Errors propagate, Quantifying error propagation	12
V	**Data Visualization** **GIS and Maps, The Visualization Process** **Visualization Strategies: Present or explore?** **The cartographic toolbox:** What kind of data do I have?, How can I map my data? **How to map?:** How to map qualitative data, How to map quantitative data, How to map the terrain elevation, How to map time series **Map Cosmetics, Map Dissemination**	12

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Principles of Geographic Information Systems-An Introductory Text Book	Editors: Otto Huisman and Rolf A.	The International Institute of Geoinformation Science and Earth Observation	Fourth	2009

2.	Principles of Geographic Information Systems	P.A Burrough and R.A.McDonnell	Oxford University Press	Third	1999
3.	Fundamentals of Spatial Information Systems,	R.Laurini and D. Thompson,	Academic Press		1994
4.	Fundamentals of Geographic Information Systems	Michael N.Demers	Wiley Publications	Fourth	2009
5.	Introduction to Geographic Information Systems	Chang Kang-tsung (Karl),	McGrawHill	Any above 3rd Edition	2013 7th Edition
6.	GIS Fundamentals: A First Text on Geographic Information Systems	Paul Bolsatd	XanEdu Publishing Inc	5th Edition	

B. Sc. (Information Technology)		Semester – VI	
Course Name: Enterprise Networking		**Course Code: USIT605 (Elective II)**	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	**Marks**
Evaluation System	**Theory Examination**	2½	75
	Internal	--	25

Unit	Details	Lectures
I	**General Network Design:** Network Design Methodology, Architectures for the Enterprise, Borderless Networks Architecture, Collaboration and Video Architecture, Data Center and Virtualization Architecture, Design Lifecycle: Plan, Build, Manage Plan Phase Build Phase Manage Phase Prepare, Plan, Design, Implement, Operate, and Optimize Phases Prepare Phase Plan Phase Design Phase Implement Phase Operate Phase Optimize Phase Summary of PPDIOO Phases Project Deliverables Design Methodology Identifying Customer Design Requirements Characterizing the Existing Network Steps in Gathering Information Network Audit Tools Network Checklist Designing the Network Topology and Solutions Top-Down Approach Pilot and Prototype Tests Design Document **Network Design Models:** Hierarchical Network Models Benefits of the Hierarchical Model, Hierarchical Network Design, Core Layer, Distribution Layer, Access Layer, Hierarchical Model Examples, Hub-and-Spoke, Design Collapsed Core, Design Enterprise Architecture Model, Enterprise Campus Module, Enterprise Edge Area, E-Commerce Module, Internet Connectivity Module, VPN/Remote Access, Enterprise WAN, Service Provider Edge Module, Remote Modules, Enterprise Branch Module, Enterprise Data Center Module, Enterprise Teleworker Module, High Availability Network Services, Workstation-to-Router Redundancy and LAN, High Availability Protocols, ARP Explicit Configuration, RDP, RIP, HSRP, VRRP, GLBP, Server Redundancy, Route Redundancy, Load Balancing, Increasing Availability, Link Media Redundancy	**12**
II	**Enterprise LAN Design:** LAN Media, Ethernet Design Rules, 100Mbps Fast Ethernet Design Rules, Gigabit Ethernet Design Rules, 1000BASE-LX Long-Wavelength Gigabit Ethernet, 1000BASE-SX Short-Wavelength Gigabit Ethernet, 1000BASE-CX Gigabit Ethernet over Coaxial Cable, 1000BASE-T Gigabit Ethernet over UTP 86, 10 Gigabit Ethernet Design Rules, 10GE Media Types, EtherChannel, Comparison of Campus Media LAN Hardware, Repeaters, Hubs, Bridges, Switches, Routers, Layer 3 Switches, Campus LAN Design and Best Practices Best Practices for Hierarchical Layers, Access Layer Best Practices, Distribution Layer Best Practices, Core Layer Best Practices, STP Design Considerations, STP Toolkit, PortFast,	**12**

	UplinkFast, BackboneFast, Loop Guard, Root Guard, BPDU Guard, BPDU Filter, VLAN and Trunk Considerations, Unidirectional Link Detection (UDLD) Protocol, Large-Building LANs, Enterprise Campus LANs, Edge Distribution, Medium-Size LANs, Small and Remote Site LANs, Server Farm Module, Server Connectivity Options, Enterprise Data Center Infrastructure, Campus LAN QoS Considerations, Multicast Traffic Considerations, CGMP, IGMP Snooping. **Data Center Design:** Enterprise DC Architecture, Data Center Foundation Components, Data Center Topology Components, Data Center Network Programmability, SDN, Controllers, APIs, ACI, Challenges in the DC, Data Center Facility Aspects, Data Center Space, Data Center Power, Data Center Cooling, Data Center Heat, Data Center Cabling, Enterprise DC Infrastructure, Data Center Storage, Data Center Reference Architecture, Defining the DC Access Layer, Defining the DC Aggregation Layer, Defining the DC Core Layer, Security in the DC, Fabric Extenders, Virtualization Overview, Challenges, Defining Virtualization and Benefits, Virtualization Risks, Types of Virtualization, Virtualization Technologies, VSS, VRF, vPC, Device Contexts, Server Virtualization, Server Scaling, Virtual Switching, Network Virtualization Design Considerations, Access Control, Path Isolation, Services Edge, Data Center Interconnect, DCI Use Cases, DCI Transport Options, DCI L2 Considerations, Load Balancing in the DC, Application Load Balancing, Network Load Balancing.	
III	**Wireless LAN Design:** Wireless LAN Technologies, WLAN Standards, ISM and UNII Frequencies, Summary of WLAN Standards, Service Set Identifier, WLAN Layer 2 Access Method, WLAN Security, Unauthorized Access, WLAN Security Design Approach, IEEE 802.1X-2001 Port-Based Authentication, Dynamic WEP Keys and LEAP, Controlling WLAN Access to Servers, WLAN Authentication, Authentication Options, WLAN Controller Components, WLC Interface Types, AP Controller Equipment Scaling, Roaming and Mobility Groups, Intracontroller Roaming, Layer 2 Intercontroller Roaming, Layer 3 Intercontroller Roaming, Mobility Groups, WLAN Design, Controller Redundancy Design: Deterministic vs. Dynamic, N+1 WLC Redundancy, N+N WLC Redundancy, N+N+1 WLC Redundancy, Radio Management and Radio Groups, RF Groups, RF Site Survey, Using EoIP Tunnels for Guest Services, Wireless Mesh for Outdoor Wireless, Mesh Design Recommendations, Campus Design Considerations, Power over Ethernet (PoE), Wireless and Quality of Service (QoS), Branch Design Considerations, Local MAC, REAP, Hybrid REAP, Branch Office Controller Options. **WAN Technologies and the Enterprise Edge:** WAN and Enterprise Edge Overview, Definition of WAN, WAN Edge Module, Enterprise	**12**

	Edge Modules, WAN Transport Technologies, ISDN, ISDN BRI Service, ISDN PRI Service, Digital Subscriber Line, Cable, Wireless, Frame Relay, Time-Division Multiplexing, Metro Ethernet, SONET/SDH, Multiprotocol Label Switching (MPLS), Dark Fiber, Dense Wavelength-Division Multiplexing, Ordering WAN Technology and Contracts, WAN and Edge Design Methodologies, Response Time, Throughput, Reliability, Bandwidth Considerations, WAN Link Categories, Optimizing Bandwidth Using QoS, Queuing, Traffic Shaping and Policing, Classification, Congestion Management, Priority Queuing, Custom Queuing, Weighted Fair Queuing, Class-Based Weighted Fair Queuing, Low-Latency Queuing, Traffic Shaping and Policing, Link Efficiency, Window Size, DMZ Connectivity, Segmenting DMZs, DMZ Services, Internet Connectivity, Centralized Internet (Branch) vs. Direct Internet (Branch), High Availability for the Internet Edge, VPN Network Design. **WAN Design** Traditional WAN Technologies Hub-and-Spoke Topology Full-Mesh Topology Partial-Mesh Topology Point-to-Point Topology Remote Site Connectivity Enterprise VPN vs. Service Provider VPN Enterprise Managed VPN: IPsec IPsec Direct Encapsulation Generic Routing Encapsulation IPsec DMVPN IPsec Virtual Tunnel Interface Design GETVPN Service Provider–Managed Offerings ,Metro Ethernet Service Provider VPNs: L2 vs. L3 ,Virtual Private Wire Services VPWS L2 VPN Considerations ,Virtual Private LAN Services VPLS L2 VPN Considerations ,MPLS, MPLS Layer 3 Design Overview MPLS L3 VPN Considerations ,VPN Benefits WAN Backup Design WAN Backup over the Internet Enterprise WAN Architecture Cisco Enterprise MAN/WAN Enterprise WAN/MAN Architecture Comparison ,Enterprise WAN Components Comparing Hardware and Software Enterprise Branch Architecture Branch Design Branch Connectivity Redundancy for Branches Single WAN Carrier vs. Dual WAN Carriers Single MPLS Carrier Site ,Dual MPLS Carriers Hybrid WAN: L3 VPN with IPsec VPN ,Internet for Branches Flat Layer 2 vs. Collapsed Core ,Enterprise Branch Profiles Small Branch Design Medium Branch Design Large Branch Design Enterprise Teleworker Design ,ISRs for Teleworkers	
IV	Internet Protocol Version 4 Design,IPv4 Header ToS IPv4 Fragmentation IPv4 Addressing ,IPv4 Address Classes Class A Addresses Class B Addresses ,Class C Addresses Class D Addresses Class E Addresses ,IPv4 Address Types IPv4 Private Addresses NAT ,IPv4 Address Subnets Mask Nomenclature IP Address Subnet Design Example Determining the Network Portion of an IP Address Variable-Length Subnet Masks, Loopback Addresses IP Telephony Networks ,IPv4 Addressing Design Goal of IPv4 Address Design , Plan for Future Use of IPv4 Addresses , Performing Route Summarization , Plan for a	**12**

Hierarchical IP Address Network , Private and Public IP Address and NAT Guidelines , Steps for Creating an IPv4 Address Plan

Case Study: IP Address Subnet Allocation , Address Assignment and Name Resolution , Recommended Practices of IP Address Assignment , BOOTP DHCP DNS , Internet Protocol Version 6 Design, IPv6 Header IPv6 Address Representation IPv4-Compatible IPv6 Addresses IPv6 Prefix Representation IPv6 Address Scope Types and Address Allocations IPv6 Address Allocations IPv6 Unicast Address Global Unicast Addresses Link-Local Addresses , Unique Local IPv6 Address Global Aggregatable IPv6 Address , IPv4-Compatible IPv6 Address IPv6 Anycast Addresses , IPv6 Multicast Addresses IPv6 Mechanisms ICMPv6 , IPv6 Neighbor Discovery Protocol IPv6 Name Resolution , Path MTU Discovery IPv6 Address-Assignment Strategies , Manual Configuration SLAAC of Link-Local Address , SLAAC of Globally Unique IPv6 Address DHCPv6 , DHCPv6 Lite IPv6 Security IPv6 Routing Protocols

RIPng OSPFv3 , BGP4 Multiprotocol Extensions (MP-BGP) for IPv6 , IPv6 Addressing Design , Planning for Addressing with IPv6 , Route Summarization with IPv6 IPv6 Private Addressing

IPv6 for the Enterprise IPv6 Address Allocation , Partly Linked IPv4 Address into IPv6, Whole IPv4 Address Linked into IPv6

IPv6 Addresses Allocated Per Location and/or Type , IPv4-to-IPv6 Transition Mechanisms and Deployment Models , Dual-Stack Mechanism IPv6 over IPv4 Tunnels , Protocol Translation Mechanisms IPv6 Deployment Models , Dual-Stack Model Hybrid Model Service Block Model ,IPv6 Deployment Model Comparison IPv6 Comparison with IPv4 ,OSPF, BGP, Route Manipulation, and IP Multicast,OSPFv2 OSPFv2 Metric OSPFv2 Adjacencies and Hello Timers , OSPFv2 Areas OSPF Area Design Considerations OSPF Router Types OSPF DRs LSA Types Autonomous System External Path Types OSPF Stub Area Types Stub Areas Totally Stubby Areas , NSSAs Virtual Links OSPFv2 Router Authentication , OSPFv2 Summary OSPFv3 OSPFv3 Changes from OSPFv2, OSPFv3 Areas and Router Types OSPFv3 LSAs OSPFv3 Summary

BGP BGP Neighbors eBGP iBGP Route Reflectors Confederations BGP Administrative Distance , BGP Attributes, Weight, and the BGP Decision Process

BGP Path Attributes Next-Hop Attribute Local Preference Attribute Origin Attribute Autonomous System Path Attribute

MED Attribute Community Attribute Atomic Aggregate and Aggregator Attributes Weight BGP Decision Process , BGP Summary , Route Manipulation PBR Route Summarization

Route Redistribution Default Metric OSPF Redistribution   Route Filtering Transit Traffic Routing Protocols on the Hierarchical Network Infrastructure IP Multicast Review , Multicast Addresses Layer 3 to Layer 2 Mapping IGMP , IGMPv1 IGMPv2 IGMPv3 CGMP IGMP Snooping , Sparse Versus Dense Multicast Multicast Source and Shared

		Trees PIM PIM-SM PIM DR Auto-RP PIMv2 Bootstrap Router , DVMRP IPv6 Multicast Addresses	
V		Managing Security Network Security Overview Security Legislation Security Threats Reconnaissance and Port Scanning Vulnerability Scanners Unauthorized Access Security Risks Targets Loss of Availability Integrity Violations and Confidentiality Breaches , Security Policy and Process Security Policy Defined , Basic Approach of a Security Policy Purpose of Security Policies, Security Policy Components Risk Assessment , Risk Index Continuous Security Integrating Security Mechanisms into Network Design Trust and Identity Management , Trust Domains of Trust Identity Passwords Tokens Certificates , Network Access Control Secure Services Encryption Fundamentals Encryption Keys VPN Protocols , Transmission Confidentiality Data Integrity Threat Defense , Physical Security Infrastructure Protection Security Management Solutions Security Solution Network Security Platforms , Trust and Identity Technologies Firewall Fundamentals , Types of Firewalls Next-Gen Firewalls NAT Placement , Firewall Guidelines Firewall ACLs , Identity and Access Control Deployments Detecting and Mitigating Threats IPS/IDS Fundamentals IPS/IDS Guidelines , Threat Detection and Mitigation Technologies , Threat-Detection and Threat-Mitigation Solutions , FirePOWER IPS Security Management Applications , Security Platform Solutions Security Management Network Integrating Security into Network Devices IOS Security , ISR G2 Security Hardware Options Securing the Enterprise , Implementing Security in the Campus Implementing Security in the Data Center Implementing Security in the Enterprise Edge Network Management Protocols, Simple Network Management Protocol SNMP Components , MIB SNMP Message Versions SNMPv1 SNMPv2 SNMPv3 , Other Network Management Technologies RMON , RMON2 NetFlow Compared to RMON and SNMP , CDP LLDP Syslog	**12**

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	CCDA200-310Official Cert Guide	ANTHONY BRUNO, CCIE No. 2738 STEVE JORDAN, CCIE No. 11293	**Cisco Press**		
2.	Network Warrior	Gary A Donabue	**O Reilly**	2nd	2011

B. Sc. (Information Technology)		Semester – VI	
Course Name: IT Services Management		**Course Code: USIT606 (Elective I)**	
Periods per week (1 Period is 50 minutes),		**5**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	2½	75
	Internal	--	25

Unit	Details	Lectures
I	**IT Service Management:** Introduction, What is service management? What are services? Business Process, Principles of Service management: Specialisation and Coordination, The agency principle, Encapsulation, Principles of systems, The service Life Cycle, Functions and processes across the life cycle. **Service Strategy Principles:** Value creation, Service Assets, Service Provider Service Structures, Service Strategy Principles. **Service Strategy:** Define the market, Develop the offerings, Develop Strategic Assets, Prepare for execution. **Challenges, Critical Success factors and risks:** Complexity, Coordination and Control, Preserving value, Effectiveness in measurement, Risks.	12
II	**Service Design:** Fundamentals, **Service Design Principles:** Goals, Balanced Design, Identifying Service requirements, identifying and documenting business requirements and drivers, Design activities, Design aspects, Subsequent design activities, Design constraints, Service oriented architecture, Business Service Management, Service Design Models **Service Design Processes:** Service Catalogue Management, Service Level Management, Capacity Management, Availability Management, IT Service Continuity Management, Information Security Management, Supplier Management **Challenges, Critical Success factors and risks:** Challenges, Risks	12
III	**Service Transition:** Fundamentals, **Service Transition Principles:** Principles Supporting Service Transition, Policies for Service Transition **Service Transition Processes:** Transition planning and support, Change Management, Service Asses Configuration Management, Service and Deployment Management, Service Validation and Testing, Evaluation, Knowledge Management. **Challenges, Critical Success factors and risks:** Challenges, Critical Success factors, Risks, Service Transition under difficult Conditions.	12
IV	**Service Operation:** Fundamentals, **Service Operation Principles:** Functions, groups, teams, departments and divisions, Achieving balance in service operations, Providing service, Operation staff involvement in service design and service transition, Operational Health, Communication, Documentation	12

		Service Operation Processes: Event Management, Incident Management, Request fulfilment, Problem Management, Access Management, Operational activities of processes covered in other lifecycle phases. Challenges, Critical Success factors and risks: Challenges, Critical Success factors, Risks	
V	Continual Service Improvement(CSI) Principles: CSI Approach, CSI and organizational change, Ownership, CSI register, External and Internal drivers, Service level management, Knowledge management, The Deming cycle, Service Measurement, IT governance, Frameworks, models, standards and quality Systems, CSI inputs and outputs. CSI Process: The seven step improvement process. CSI Methods nad Techniques: Methods and techniques, Assessments, benchmarking, Service Measurement, Metrics, Return on Investment, Service reporting, CSI and other service management processes, Organising for CSI: Organisational development, Functions, roles, Customer Engagement, Responsibility model - RACI, Competence and training. Technology considerations: Tools to support CSI activities. Implementing CSI: Critical Considerations for implementing CSI, The start, Governance, CSI and organisational change, Communication Strategy and Plan	12	

Books and References:					
Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	ITIL v3 Foundation Complete Certification Kit				2009
2.	ITIL v3 Service Strategy		OGC/TSO		
3.	ITIL v3 Service Transition		OGC/TSO		
4.	ITIL v3 Service Operation		OGC/TSO		
5.	ITIL Continual Service Improvement		TSO	2011	2011

B. Sc. (Information Technology)		Semester – VI	
Course Name: Cyber Laws		**Course Code: USIT607 (Elective I)**	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	**Marks**
Evaluation System	**Theory Examination**	2½	75
	Internal	--	25

Unit	Details	Lectures
I	**Power of Arrest Without Warrant Under the IT Act, 2000:** A Critique, Crimes of this Millennium, Section 80 of the IT Act, 2000 – A Weapon or a Farce? Forgetting the Line Between Cognizable and Non-Cognizable Offences, Necessity of Arrest without Warrant from Any Place, Public or Otherwise, Check and Balances Against Arbitrary Arrests, Arrest for "About to Commit" an Offence Under the IT Act: A Tribute to Draco, Arrest, But NO Punishment! **Cyber Crime and Criminal Justice: Penalties, Adjudication and Appeals Under the IT Act, 2000:** Concept of "Cyber Crime " and the IT Act , 2000, Hacking, Teenage Web Vandals, Cyber Fraud and Cyber Cheating, Virus on the Internet, Defamation, Harassment and E-mail Abuse, Cyber Pornography, Other IT Act Offences, Monetary Penalties, Adjudication and Appeals Under IT Act , 2000, Network Service Providers, Jurisdiction and Cyber Crime, Nature of Cyber Criminality, Strategies to Tackle Cyber Crime and Trends, Criminal Justice in India and Implications on Cyber Crime.	12
II	**Contracts in the Infotech World:** Contracts in the Infotech World, Click-Wrap and Shrink-Wrap Contract: Status under the Indian Contract Act, 1872, Contract Formation Under the Indian Contract Act, 1872, Contract Formation on the Internet, Terms and Conditions of Contracts. **Jurisdiction in the Cyber World:** Questioning the Jurisdiction and Validity of the Present Law of Jurisdiction, Civil Law of Jurisdiction in India, Cause of Action, Jurisdiction and the Information Technology Act,2000, Foreign Judgements in India, Place of Cause of Action in Contractual and IPR Disputes, Exclusion Clauses in Contracts, Abuse of Exclusion Clauses, Objection of Lack of Jurisdiction, Misuse of the Law of Jurisdiction, Legal Principles on Jurisdiction in the United State of America, Jurisdiction Disputes w.r.t. the Internet in the United State of America.	12
III	**Battling Cyber Squatters and Copyright Protection in the Cyber World:** Concept of Domain Name and Reply to Cyber Squatters, Meta-Tagging, Legislative and Other Innovative Moves Against Cyber Squatting, The Battle Between Freedom and Control on the Internet, Works in Which Copyright Subsists and meaning of Copyright, Copyright Ownership and Assignment, License of Copyright, Copyright Terms and Respect for Foreign Works, Copyright	12

	Infringement, Remedies and Offences, Copyright Protection of Content on the Internet; Copyright Notice, Disclaimer and Acknowledgement, Downloading for Viewing Content on the Internet, Hyper-Linking and Framing, Liability of ISPs for Copyright Violation in the Cyber World: Legal Developments in the US, Napster and its Cousins: A Revolution on the Internet but a Crisis for Copyright Owners, Computer Software Piracy.	
IV	**E-Commerce Taxation: Real Problems in the Virtual World:** A Tug of War on the Concept of 'Permanent Establishment', Finding the PE in Cross Border E-Commerce, The United Nations Model Tax Treaty, The Law of Double Taxation Avoidance Agreements and Taxable Jurisdiction Over Non-Residents, Under the Income Tax Act, 1961, Tax Agents of Non-Residents under the Income Tax Act,1961 and the Relevance to E-Commerce, Source versus Residence and Classification between Business Income and Royalty, The Impact of the Internet on Customer Duties, Taxation Policies in India: At a Glance. **Digital Signature, Certifying Authorities and E-Governance:** Digital Signatures, Digital Signature Certificate, Certifying Authorities and Liability in the Event of Digital Signature Compromise, E-Governance in India: A Warning to Babudom!	**12**
V	**The Indian Evidence Act of 1872 v. Information Technology Act, 2000:** Status of Electronic Records as Evidence, Proof and Management of Electronic Records; Relevancy, Admissibility and Probative Value of E-Evidence, Proving Digital Signatures, Proof of Electronic Agreements, Proving Electronic Messages, Other Amendments in the Indian Evidence Act by the IT Act, Amendments to the Bankers Books Evidence Act, 1891 and Reserve Bank of India Act, 1934. **Protection of Cyber Consumers in India:** Are Cyber Consumers Covered Under the Consumer Protection Act? Goods and Services, Consumer Complaint, Defect in Goods and Deficiency in Services, Restrictive and Unfair Trade Practices, Instances of Unfair Trade Practices, Reliefs Under CPA, Beware Consumers, Consumer Foras, Jurisdiction and Implications on cyber Consumers in India, Applicability of CPA to Manufacturers, Distributors, Retailers and Service Providers Based in Foreign Lands Whose Goods are Sold or Services Provided to a Consumer in India. Amendments in Indian IT Act 2000	**12**

Books and References:					
Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Cyber Law Simplified	Vivek Sood	TMH Education		2001
2.	Cybersecurity Law	Jeff Kosseff	Wiley		2017

B. Sc. (Information Technology)		Semester – VI	
Course Name: Project Implementation		**Course Code: USIT6P1**	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	**Marks**
Evaluation System	**Practical Examination**	2½	50
	Internal	--	-

The details are given in Appendix – I

B. Sc. (Information Technology)		Semester – VI	
Course Name: Security in Computing Practical		Course Code: USIT6P2	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2½	50
	Internal	--	-

Practical No	Details
1	**Configure Routers**
a	OSPF MD5 authentication.
b	NTP.
c	to log messages to the syslog server.
d	to support SSH connections.
2	**Configure AAA Authentication**
a	Configure a local user account on Router and configure authenticate on the console and vty lines using local AAA
b	Verify local AAA authentication from the Router console and the PC-A client
3	**Configuring Extended ACLs**
a	**Configure, Apply and Verify an Extended Numbered ACL**
4	**Configure IP ACLs to Mitigate Attacks and IPV6 ACLs**
a	Verify connectivity among devices before firewall configuration.
b	Use ACLs to ensure remote access to the routers is available only from management station PC-C.
c	Configure ACLs on to mitigate attacks.
d	Configuring IPv6 ACLs
5	**Configuring a Zone-Based Policy Firewall**
6	Configure IOS Intrusion Prevention System (IPS) Using the CLI
a	Enable IOS IPS.
b	Modify an IPS signature.
7	**Layer 2 Security**
a	Assign the Central switch as the root bridge.
b	Secure spanning-tree parameters to prevent STP manipulation attacks.
c	Enable port security to prevent CAM table overflow attacks.
8	**Layer 2 VLAN Security**
9	**Configure and Verify a Site-to-Site IPsec VPN Using CLI**

10	**Configuring ASA Basic Settings and Firewall Using CLI**
a	Configure basic ASA settings and interface security levels using CLI
b	Configure routing, address translation, and inspection policy using CLI
c	Configure DHCP, AAA, and SSH
d	Configure a DMZ, Static NAT, and ACLs

B. Sc. (Information Technology)		Semester – VI	
Course Name: Business Intelligence Practical		**Course Code: USIT6P3**	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	**Marks**
Evaluation System	**Practical Examination**	2½	50
	Internal	--	-

Practical No	Details
1	Import the legacy data from different sources such as (Excel , SqlServer, Oracle etc.) and load in the target system. (You can download sample database such as Adventureworks, Northwind, foodmart etc.)
2	Perform the Extraction Transformation and Loading (ETL) process to construct the database in the Sqlserver.
3	a. Create the Data staging area for the selected database. b. Create the cube with suitable dimension and fact tables based on ROLAP, MOLAP and HOLAP model.
4	a.Create the ETL map and setup the schedule for execution. b. Execute the MDX queries to extract the data from the datawarehouse.
5	a. Import the datawarehouse data in Microsoft Excel and create the Pivot table and Pivot Chart. b. Import the cube in Microsoft Excel and create the Pivot table and Pivot Chart to perform data analysis.
6	Apply the what – if Analysis for data visualization. Design and generate necessary reports based on the data warehouse data.
7	Perform the data classification using classification algorithm.
8	Perform the data clustering using clustering algorithm.
9	Perform the Linear regression on the given data warehouse data.
10	Perform the logistic regression on the given data warehouse data.

The BI tools such as Tableau / Power BI / BIRT / R / Excel or any other can be used.

B. Sc. (Information Technology)		Semester – VI	
Course Name: Principles of Geographical Information System Practical		**Course Code: USIT6P4 (Elective II)**	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	**Marks**
Evaluation System	**Practical Examination**	2½	50
	Internal	--	-

Practical No	Details
0	Familiarizing Quantum GIS: Installation of QGIS, datasets for both Vector and Raster data, Maps.
1	Creating and Managing Vector Data: Adding vector layers, setting properties, formatting, calculating line lengths and statistics
2	Exploring and Managing Raster data: Adding raster layers, raster styling and analysis, raster mosaicking and clipping
3	Making a Map, Working with Attributes, Importing Spreadsheets or CSV files Using Plugins, Searching and Downloading OpenStreetMap Data
4	Working with attributes, terrain Data
5	Working with Projections and WMS Data
6	Georeferencing Topo Sheets and Scanned Maps Georeferencing Aerial Imagery Digitizing Map Data
7	Managing Data Tables and Saptial data Sets: Table joins, spatial joins, points in polygon analysis, performing spatial queries
8	Advanced GIS Operations 1: Nearest Neighbor Analysis, Sampling Raster Data using Points or Polygons, Interpolating Point Data
9	Advance GIS Operations 2: Batch Processing using Processing Framework Automating Complex Workflows using Processing Modeler Automating Map Creation with Print Composer Atlas
10	Validating Map data

B. Sc. (Information Technology)		Semester – VI	
Course Name: Advanced Networking Practical		**Course Code: USIT6P5 (Elective II)**	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	**Marks**
Evaluation System	**Practical Examination**	2½	50
	Internal	--	-

Practical No	Details
1	**Configuring OSPF – I**
a	Single-Area OSPF Link Costs and Interface Priorities
b	Multi-Area OSPF with Stub Areas and Authentication
2	**Configuring OSPF – II**
a	OSPF Virtual Links and Area Summarization
b	OSPF over Frame Relay
3	**Redistribution and Administrative Distances**
a	Redistribution Between RIP and OSPF
b	Manipulating Administrative Distances
4	**BGP**
a	Configuring BGP with Default Routing
b	Using the AS_PATH Attribute
c	BGP Route Reflectors and Route Filters
5	**IPv6**
a	Configuring OSPF for IPv6
b	Configuring 6to4 Tunnels
6	**VLANs and EtherChannel**
a	Static VLANS, VLAN Trunking, and VTP Domains and Modes
b	Configuring EtherChannel
7	**Spanning Tree Protocol**
a	Spanning Tree Protocol (STP) Default Behavior
b	Modifying Default Spanning Tree Behavior
8	**VLAN and Spanning Tree**
a	Per-VLAN Spanning Tree Behavior
b	Multiple Spanning Tree

9	**Internal VLAN Routing**
a	Inter-VLAN Routing with an External Router
b	Inter-VLAN Routing with an Internal Route Processor
10	**Configure NAT Services**

B. Sc. (Information Technology)		Semester – VI	
Course Name: Advanced Mobile Programming Practical		**Course Code: USIT6P6**	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2½	50
	Internal	--	--

Practical No	Details
1	**Introduction to Android, Introduction to Android Studio IDE, Application Fundamentals:** Creating a Project, Android Components, Activities, Services, Content Providers, Broadcast Receivers, Interface overview, Creating Android Virtual device, USB debugging mode, Android Application Overview. Simple "Hello World" program.
2	**Programming Resources** Android Resources: (Color, Theme, String, Drawable, Dimension, Image),
3	**Programming Activities and fragments** Activity Life Cycle, Activity methods, Multiple Activities, Life Cycle of fragments and multiple fragments.
4	**Programs related to different Layouts** Coordinate, Linear, Relative, Table, Absolute, Frame, List View, Grid View.
5	**Programming UI elements** AppBar, Fragments, UI Components
6	**Programming menus, dialog, dialog fragments**
7	**Programs on Intents, Events, Listeners and Adapters** The Android Intent Class, Using Events and Event Listeners
8	**Programs on Services, notification and broadcast receivers**
9	**Database Programming with SQLite**
10	**Programming threads, handles and asynchronized programs**
11	**Programming Media API and Telephone API**
12	**Programming Security and permissions**
13	**Programming Network Communications and Services (JSON)**

APPENDIX – 1

Project Dissertation Semester V and Project Implementation Semester VI

Chapter 1 to 4 should be submitted in Semester V in spiral binding. These chapter have also to be included in Semester VI report. Semester VI report has to be hard bound with golden embossing. Students will be evaluated based on the dissertation in semester V and dissertation and viva voce in Semester VI.

I. OBJECTIVES

- Describe the Systems Development Life Cycle (SDLC).

- Evaluate systems requirements.

- Complete a problem definition.

- Evaluate a problem definition.

- Determine how to collect information to determine requirements.

- Perform and evaluate feasibility studies like cost-benefit analysis, technical feasibility, time feasibility and Operational feasibility for the project.

- Work on data collection methods for fact finding.

- Construct and evaluate data flow diagrams.

- Construct and evaluate data dictionaries.

- Evaluate methods of process description to include structured English, decision tables and decision trees.

- Evaluate alternative tools for the analysis process.

- Create and evaluate such alternative graphical tools as systems flow charts and state transition diagrams.

- Decide the S/W requirement specifications and H/W requirement specifications.

- Plan the systems design phase of the SDLC.

- Distinguish between logical and physical design requirements.

- Design and evaluate system outputs.

- Design and evaluate systems inputs.

- Design and evaluate validity checks for input data.

- Design and evaluate user interfaces for input.

- Design and evaluate file structures to include the use of indexes.

- Estimate storage requirements.

- Explain the various file update processes based on the standard file organizations.

- Decide various data structures.

- Construct and evaluate entity-relationship (ER) diagrams for RDBMS related projects.

- Perform normalization for the unnormalized tables for RDBMS related projects

- Decide the various processing systems to include distributed, client/server, online and others.

- Perform project cost estimates using various techniques.

- Schedule projects using both GANTT and PERT charts.

- Perform coding for the project.

- Documentation requirements and prepare and evaluate systems documentation.

- Perform various systems testing techniques/strategies to include the phases of testing.

- Systems implementation and its key problems.

- Generate various reports.

- Be able to prepare and evaluate a final report.

- Brief the maintenance procedures and the role of configuration management in operations.

- To decide the future scope and further enhancement of the system.

- Plan for several appendices to be placed in support with the project report documentation.

- Decide the various processing systems to include distributed, client/server, online and others.

- Perform project cost estimates using various techniques.

- Schedule projects using both GANTT and PERT charts.

- Perform coding for the project.

- Documentation requirements and prepare and evaluate systems documentation.

- Perform various systems testing techniques/strategies to include the phases of testing.

- Systems implementation and its key problems.

- Generate various reports.

- Be able to prepare and evaluate a final report.

- Brief the maintenance procedures and the role of configuration management in operations.

- To decide the future scope and further enhancement of the system.

- Plan for several appendices to be placed in support with the project report documentation.

- Work effectively as an individual or as a team member to produce correct, efficient, well-organized and documented programs in a reasonable time.
- Recognize problems that are amenable to computer solutions, and knowledge of the tool necessary for solving such problems.
- Develop of the ability to assess the implications of work performed.
- Get good exposure and command in one or more application areas and on the software
- Develop quality software using the software engineering principles
- Develop of the ability to communicate effectively.

II. Type of the Project

The majority of the students are expected to work on a real-life project preferably in some industry/ Research and Development Laboratories/Educational Institution/Software Company. Students are encouraged to work in the areas listed below. However, it is *not mandatory* for a student to work on a real-life project. The student can formulate a project problem with the help of her/his Guide and submit the project proposal of the same. **Approval of the project proposal is mandatory**. If approved, the student can commence working on it, and complete it. Use the latest versions of the software packages for the development of the project.

III. SOFTWARE AND BROAD AREAS OF APPLICATION

FRONT END / GUI Tools	.Net Technologies,Java
DBMS/BACK END	Oracle, SQL Plus, MY SQL, SQL Server,
LANGUAGES	C, C++, Java, VC++, C#, R,Python
SCRIPTING LANGUAGES	PHP,JSP, SHELL Scripts (Unix), TcL/TK,
.NET Platform	F#,C#. Net, Visual C#. Net, ASP.Net
MIDDLE WARE (COMPONENT) TECHNOLOGIES	COM/DCOM, Active-X, EJB
UNIX INTERNALS	Device Drivers, RPC, Threads, Socket programming
NETWORK/WIRELESS TECHNOLOGIES	-

REALTIME OPERATING SYSTEM/ EMBEDDED SKILLS	LINUX, Raspberry Pi, Arduino, 8051
APPLICATION AREAS	Financial / Insurance / Manufacturing / Multimedia / Computer Graphics / Instructional Design/ Database Management System/ Internet / Intranet / Computer Networking-Communication Software development/ E-Commerce/ ERP/ MRP/ TCP-IP programming / Routing protocols programming/ Socket programming.

IV. Introduction

The project report should be documented with scientific approach to the solution of the problem that the students have sought to address. The project report should be prepared in order to solve the problem in a methodical and professional manner, making due references to appropriate techniques, technologies and professional standards. The student should start the documentation process from the first phase of software development so that one can easily identify the issues to be focused upon in the ultimate project report. The student should also include the details from the project diary, in which they will record the progress of their project throughout the course. The project report should contain enough details to enable examiners to evaluate the work. The important points should be highlighted in the body of the report, with details often referred to appendices.

1.1 PROJECT REPORT:

Title Page

Original Copy of the Approved Proforma of the Project Proposal

Certificate of Authenticated work

Role and Responsibility Form

Abstract

Acknowledgement

Table of Contents

Table of Figures

CHAPTER 1: INTRODUCTION

1.1 Background

1.2 Objectives

1.3 Purpose, Scope, and Applicability

The documentation should use tools like star UML, Visuo for windows, Rational Rose for design as part of Software Project Management Practical Course. The documentation should be spiral bound for semester V and the entire documentation should be hard bound during semester VI.

5.2.1 Code Efficiency

5.3 Testing Approach

5.3.1 Unit Testing

5.3.2 Integrated Testing

5.3.3 Beta Testing

5.4 Modifications and Improvements

5.5 Test Cases

CHAPTER 6: RESULTS AND DISCUSSION

6.1 Test Reports

6.2 User Documentation

CHAPTER 7: CONCLUSIONS

7.1 Conclusion

7.1.1 Significance of the System

7.2 Limitations of the System

7.3 Future Scope of the Project

REFERENCES

GLOSSARY

APPENDIX A

APPENDIX B

V. EXPLANATION OF CONTENTS

Title Page

Sample format of Title page is given in Appendix 1 of this block. Students should follow the given format.

Original Copy of the Approved Proforma of the Project Proposal

Sample Proforma of Project Proposal is given in Appendix 2 of this block. Students should follow the given format.

Certificate of Authenticated work

Sample format of Certificate of Authenticated work is given in Appendix 3 of this block. Students should follow the given format.

Role and Responsibility Form

Sample format for Role and Responsibility Form is given in Appendix 4 of this block.

Students should follow the given format.

Abstract

This should be one/two short paragraphs (100-150 words total), summarising the project work. It is important that this is not just a re-statement of the original project outline. A suggested flow is background, project aims and main achievements. From the abstract, a reader should be able to ascertain if the project is of interest to them and, it should present results of which they may wish to know more details.

Acknowledgements

This should express student's gratitude to those who have helped in the preparation of project.

Table of Contents: The table of contents gives the readers a view of the detailed structure of the report. The students would need to provide section and subsection headings with associated pages. The formatting details of these sections and subsections are given below.

Table of Figures: List of all Figures, Tables, Graphs, Charts etc. along with their page numbers in a table of figures.

Chapter 1: Introduction

The introduction has several parts as given below:

Background: A description of the background and context of the project and its relation to work already done in the area. Summarise existing work in the area concerned with the project work.

Objectives: Concise statement of the aims and objectives of the project. Define exactly what is going to be done in the project; the objectives should be about 30 /40 words.

Purpose, Scope and Applicability: The description of Purpose, Scope, and Applicability are given below:

• Purpose: Description of the topic of the project that answers questions on why this project is being done. How the project could improve the system its significance and theoretical framework.

• Scope: A brief overview of the methodology, assumptions and limitations. The students should answer the question: What are the main issues being covered in the project? What are the main functions of the project?

• Applicability: The student should explain the direct and indirect applications of their work. Briefly discuss how this project will serve the computer world and people.

Achievements: Explain what knowledge the student achieved after the completion of the work. What contributions has the project made to the chosen area? Goals achieved - describes the degree to which the findings support the original objectives laid out by the project. The goals may be partially or fully achieved, or exceeded.

Organisation of Report: Summarising the remaining chapters of the project report, in effect, giving the reader an overview of what is to come in the project report.

Chapter 2: Survey of Technologies

In this chapter Survey of Technologies should demonstrate the students awareness and understanding of Available Technologies related to the topic of the project. The student should give the detail of all the related technologies that are necessary to complete the project. The should describe the technologies available in the chosen area and present a comparative study of all those Available Technologies. Explain why the student selected the one technology for the completion of the objectives of the project.

Chapter 3: Requirements and Analysis

Problem Definition: Define the problem on which the students are working in the project.

Provide details of the overall problem and then divide the problem in to sub-problems. Define each sub-problem clearly.

Requirements Specification: In this phase the student should define the requirements of the system, independent of how these requirements will be accomplished. The Requirements Specification describes the things in the system and the actions that can be done on these things. Identify the operation and problems of the existing system.

Planning and Scheduling: Planning and scheduling is a complicated part of software development. Planning, for our purposes, can be thought of as determining all the small tasks that must be carried out in order to accomplish the goal. Planning also takes into account, rules, known as constraints, which, control when certain tasks can or cannot happen. Scheduling can be thought of as determining whether adequate resources are available to carry out the plan. The student should show the Gantt chart and Program Evaluation Review Technique (PERT).

Software and Hardware Requirements: Define the details of all the software and hardware needed for the development and implementation of the project.

• Hardware Requirement: In this section, the equipment, graphics card, numeric co-processor, mouse, disk capacity, RAM capacity etc. necessary to run the software must be noted.
• Software Requirements: In this section, the operating system, the compiler, testing tools, linker, and the libraries etc. necessary to compile, link and install the software must be listed.

Preliminary Product Description: Identify the requirements and objectives of the new system. Define the functions and operation of the application/system the students are developing as project.

Conceptual Models: The student should understand the problem domain and produce a model of the system, which describes operations that can be performed on the system, and the allowable sequences of those operations. Conceptual Models could consist of complete Data Flow Diagrams, ER diagrams, Object-oriented diagrams, System Flowcharts etc.

Chapter 4: System Design

Describes desired features and operations in detail, including screen layouts, business rules, process diagrams, pseudocode and other documentation.

Basic Modules: The students should follow the divide and conquer theory, so divide the overall problem into more manageable parts and develop each part or module separately. When all modules are ready, the student should integrate all the modules into one system. In this phase, the student should briefly describe all the modules and the functionality of these modules.

Data Design: Data design will consist of how data is organised, managed and manipulated.

• Schema Design: Define the structure and explanation of schemas used in the project.

• Data Integrity and Constraints: Define and explain all the validity checks and constraints provided to maintain data integrity.

Procedural Design: Procedural design is a systematic way for developing algorithms or procedurals.

• Logic Diagrams: Define the systematical flow of procedure that improves its comprehension and helps the programmer during implementation. e.g., Control Flow Chart, Process Diagrams etc.

• Data Structures: Create and define the data structure used in procedures.

• Algorithms Design: With proper explanations of input data, output data, logic of processes, design and explain the working of algorithms.

User Interface Design: Define user, task, environment analysis and how to map those requirements in order to develop a "User Interface". Describe the external and internal components and the architecture of user interface. Show some rough pictorial views of the user interface and its components.

Security Issues: Discuss Real-time considerations and Security issues related to the project and explain how the student intends avoiding those security problems. What are the security policy plans and architecture?

Test Cases Design: Define test cases, which will provide easy detection of errors and mistakes with in a minimum period of time and with the least effort. Explain the different conditions in which the students wish to ensure the correct working of the project.

Chapter 5: Implementation and Testing

Implementation Approaches: Define the plan of implementation, and the standards the students have used in the implementation.

Coding Details and Code Efficiency: Students not need include full source code, instead, include only the important codes (algorithms, applets code, forms code etc). The program code should contain comments needed for explaining the work a piece of code does. Comments may be needed to explain why it does it, or, why it does a particular way.

The student can explain the function of the code with a shot of the output screen of that program code.

• Code Efficiency: The student should explain how the code is efficient and how the students have handled code optimisation.

Testing Approach: Testing should be according to the scheme presented in the system design chapter and should follow some suitable model – e.g., category partition, state machine-based. Both functional testing and user-acceptance testing are appropriate. Explain the approach of testing.

• Unit Testing: Unit testing deals with testing a unit or module as a whole. This would test the interaction of many functions but, do confine the test within one module.

• Integrated Testing: Brings all the modules together into a special testing environment, then checks for errors, bugs and interoperability. It deals with tests for the entire application. Application limits and features are tested here.

Modifications and Improvements: Once the students finish the testing they are bound to be faced with bugs, errors and they will need to modify your source code to improve the system. Define what modification are implemented in the system and how it improved the system.

Chapter 6: Results and Discussion

Test Reports: Explain the test results and reports based on the test cases, which should show that the project is capable of facing any problematic situation and that it works fine in different conditions. Take the different sample inputs and show the outputs.

User Documentation: Define the working of the software; explain its different functions, components with screen shots. The user document should provide all the details of the product in such a way that any user reading the manual, is able to understand the working and functionality of the document.

Chapter 7: Conclusions

Conclusion: The conclusions can be summarised in a fairly short chapter (2 or 3 pages). This chapter brings together many of the points that would have made in the other chapters.

Limitations of the System: Explain the limitations encountered during the testing of the project that the students were not able to modify. List the criticisms accepted during the demonstrations of the project.

Future Scope of the Project describes two things: firstly, new areas of investigation prompted by developments in this project, and secondly, parts of the current work that was not completed due to time constraints and/or problems encountered.

REFERENCES

It is very important that the students acknowledge the work of others that they have used or adapted in their own work, or that provides the essential background or context to the project. The use of references is the standard way to do this. Please follow the given standard for the references for books, journals, and online material. The citation is mandatory in both the reports.

E.g:

Linhares, A., & Brum, P. (2007). Understanding our understanding of strategic scenarios: What role do chunks play? *Cognitive Science*, *31*(6), 989-1007. https://doi.org/doi:10.1080/03640210701703725

Lipson, Charles (2011). Cite right : A quick guide to citation styles; MLA, APA, Chicago, the sciences, professions, and more (2nd ed.). Chicago [u.a.]: University of Chicago Press. p. 187. ISBN 9780226484648.

Elaine Ritchie, J Knite. (2001). *Artificial Intelligence, Chapter 2 , p.p 23 - 44.* Tata McGrawHill.

GLOSSARY

If you the students any acronyms, abbreviations, symbols, or uncommon terms in the project report then their meaning should be explained where they first occur. If they go on to use any of them extensively then it is helpful to list them in this section and define the meaning.

APPENDICES

These may be provided to include further details of results, mathematical derivations, certain illustrative parts of the program code (e.g., class interfaces), user documentation etc.

In particular, if there are technical details of the work done that might be useful to others who wish to build on this work, but that are not sufficiently important to the project as a whole to justify being discussed in the main body of the project, then they should be included as appendices.

VI. SUMMARY

Project development usually involves an engineering approach to the design and development of a software system that fulfils a practical need. Projects also often form an important focus for discussion at interviews with future employers as they provide a detailed example of what the students are capable of achieving. In this course the students can choose your project topic from the lists given in Unit 4: Category-wise Problem Definition.

VII. FURTHER READINGS

1. Modern Systems Analysis and Design; Jeffrey A. Hoffer, Joey F. George, Joseph,S. Valacich; Pearson Education; Third Edition; 2002.
2. ISO/IEC 12207: Software Life Cycle Process
   (http://www.software.org/quagmire/descriptions/iso-iec12207.asp).
3. IEEE 1063: Software User Documentation (http://ieeexplore.ieee.org).
4. ISO/IEC: 18019: Guidelines for the Design and Preparation of User Documentation for Application Software.
5. http://www.sce.carleton.ca/squall.
6. http://en.tldp.org/HOWTO/Software-Release-Practice-HOWTO/documentation.html.
7. http://www.sei.cmu.edu/cmm/

PROFORMA FOR THE APPROVAL PROJECT PROPOSAL

(Note: All entries of the proforma of approval should be filled up with appropriate and complete information. Incomplete proforma of approval in any respect will be summarily rejected.)

PNR **No.: ……………………**   Roll no**: _____**

1. Name of the Student

    _____

2. Title of the Project

    _____

3. Name of the Guide

    _____

4. Teaching experience of the Guide _____

5.   Is this your first submission?   Yes ☐   No ☐

Signature of the Student   Signature of the Guide

Date: …………………   Date: ……………………….

Signature of the Coordinator

Date: …………………

TITLE OF THE PROJECT
(NOT EXCEEDING 2 LINES, 24 BOLD, ALL CAPS)

A Project Report (12 Bold)
Submitted in partial fulfillment of the
Requirements for the award of the Degree of (size-12)

BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY)(14 BOLD, CAPS)

By(12 Bold)

Name of The Student (size-15, title case)
Seat Number (size-15)

Under the esteemed guidance of (13 bold)
Mr./Mrs. Name of The Guide (15 bold, title case)
Designation (14 Bold, title case)

COLLEGE LOGO

DEPARTMENT OF INFORMATION TECHNOLOGY(12 BOLD, CAPS)
COLLEGE NAME (14 BOLD, CAPS)
(Affiliated to University of Mumbai) (12, Title case, bold, italic)
CITY, PIN CODE(12 bold, CAPS)
MAHARASHTRA (12 bold, CAPS)
YEAR (12 bold)

COMPANY CERTIFICATE (if applicable)

Abstract (20bold, caps, centered)

Content (12, justified)

Note: Entire document should be with <u>1.5</u> <u>line spacing and all paragraphs should start with 1 tab space.</u>

ACKNOWLEDGEMENT
(20, BOLD, ALL CAPS, CENTERED)

The acknowledgement should be in times new roman,   12 font with 1.5 line spacing, justified.

(Declaration page format)

DECLARATION (20 bold, centered, allcaps)

Content (12, justified)

I here by declare that the project entitled, "**Title of the Project**" done at **place where the project is done**, has not been in any case duplicated to submit to any other university for the award of any degree. To the best of my knowledge other than me, no one has submitted to any other university.

The project is done in partial fulfillment of the requirements for the award of degree of **BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY)** to be submitted as final semester project as part of our curriculum.

Name and Signature of the Student

TABLE OF CONTENTS (20bold, caps, centered)

Should be generated automatically using word processing software.

List of Tables (20 bold, centered, Title Case)

Should be generated automatically using word processing software.

List of Figures (20 bold, centered, Title Case)

Should be generated automatically using word processing software.

Chapter 1

Introduction (20 Bold, centered)

Content or text (12, justified)

Note: Introduction has to cover brief description of the project with minimum 4 pages.

Chapter 2

System Analysis (20 bold, Centered)

Subheadings are as shown below with following format (16 bold, CAPS)

2.1 Existing System (16 Bold)

2.1.1 ------------ (14 bold, title case)

2.1.1.1 ------------ (12 bold, title case)

2.2 Proposed System

2.3 Requirement Analysis

2.4 Hardware Requirements

2.5 Software Requirements

2.6 Justification of Platform – (how h/w & s/w satisfying the project)

Table 2.1: Caption

Chapter 3

System Design (20 bold, centered)

Subheadings are as shown below with following format (16 bold, CAPS)
Specify figures as Fig 11.1 – caption

3.1   Module Division
3.2   Data Dictionary
3.3   E-R Diagrams
3.4   Data Flow Diagrams / UML

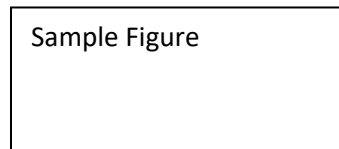Note: write brief description at the bottom of all diagrams

Sample Figure

Fig. 3.1: Caption

Chapter 4

Implementation and Testing (20 bold, centered)

4.1 Code (Place Core segments)
Content includes description about coding phase in your project (Font-12)
(* don't include complete code-----just description)

4.2 Testing Approach
Subheadings are as shown below with following format (16 bold, CAPS)

4.2.1 Unit Testing
4.2.2 Integration Testing

Note:

- ➢ Explain about above testing methods
- ➢ Explain how the above techniques are applied in your project
    Provide Test plans, test cases, etc relevant to your project

Chapter 5

Results and Discussions (20 bold, centered)

Note: Place Screen Shots and write the functionality of each screen at the bottom

Chapter 6

Conclusion and Future Work (20 bold, centered)

The conclusions can be summarized in a fairly short chapter around 300 words. Also include limitations of your system and future scope (12, justified)

Chapter 7

References (20 bold, centered)

Content (12, LEFT)

[1] Title of the book, Author

[2] Full URL of online references

[3] ------------

* NOTE ABOUT PROJECT VIVA VOCE:

Student may be asked to write code for problem during VIVA to demonstrate his coding capabilities and he/she may be asked to write any segment of coding used in the in the project. The project can be done in group of at most four students. However, the length and depth of the project should be justified for the projects done in group. A big project can be modularised and different modules can be assigned as separate project to different students.

Marks Distribution:

Semester V: 50 Marks

Documentation: 50 marks

Semester VI: 150 Marks

Documentation: 50 Marks:

Implementation and Viva Voce: 100 Marks

The plagiarism should be maintained as per the UGC guidelines.