The Kelkar Education Trust's

V. G. Vaze College of Arts, Science and Commerce

(Autonomous)

Syllabus for F. Y B.Sc. – I.T.

(June 2024 Onwards)

Programme: B.Sc.

Subject : Information Technology Semester

I & II

Semester I			
Code	**Course Type**	**Course Title**	**Credits**
VGVUSTMPP101	Major Subject	Principles of Programming Languages using C	2
VGVUSTMMA101	Major Subject	Microprocessor Architecture with 8085	2
VGVUSTMP101	Major Subject Practical	Principles of Programming Languages using C and Microprocessor Architecture with 8085 Practical	2
VGVUOE112	Open Elective Subject	Basic Accounting and Practices	4
VGVUOE115		Financial Markets	
VGVUOE119		History of Media	
VGVUOE120		Indian Financial System	
VGVUOE121		Introduction to Managerial Practices	
VGVUOE122		Organizational Behaviour	
VGVUOE123		The India Story	
VGVUOE124		Visual Communication	
VGVUSTVSE101	Vocational Skill Enhancement Course	Discrete Mathematics	2
VGVUSTVSEP101	Vocational Skill Enhancement Course Pr	Numerical Computations using Scilab Practical	2
VGVUFAE103	Ability Enhancement Course	English	2
VGVUVE108	Value Education Course	Environmental study for sustainable IT - I	2
VGVUIKS106	Indian Knowledge System	Ancient Vedic Mathematics	2
VGVUCC101	Co-Curricular Courses	Community Engagement Activities	2
VGVUCC102		Cultural Activities	
VGVUCC103		National Service Scheme (NSS)	
VGVUCC104		Sports Activities	
VGVUCC105		Yoga	
		Total Credits	**22**

Semester II			
Code	**Course Type**	**Course Title**	**Credits**
VGVUSTMOP201	Major Subject	Object Oriented Programming using C++	2
VGVUSTMDS201	Major Subject	Database Management Systems	2
VGVUSTMP201	Major Subject Practical	Object Oriented Programming using C++ and Database Management Systems Practical	2
VGVUSTNDS201	Minor Subject	Basics of Data Science	2
VGVUOE212	Open Elective Subject	Basic Accounting and Practices	4
VGVUOE215		Financial Markets	
VGVUOE219		History of Media	
VGVUOE220		Indian Financial System	
VGVUOE221		Introduction to Managerial Practices	
VGVUOE222		Organizational Behaviour	
VGVUOE223		The India Story	
VGVUOE224		Visual Communication	
VGVUSTVSE201	Vocational Skill Enhancement Course	Fundamentals of Digital Electronics	2
VGVUSTVSEP201	Vocational Skill Enhancement Course Pr	Fundamentals of Digital Electronics Practical	2
VGVUFAE204	Ability Enhancement Course	Effective Communication Skills	2
VGVUVE206	Value Education Course	Environmental study for sustainable IT - II	2
VGVUCC201	Co-Curricular Courses	Community Engagement Activities	2
VGVUCC202		Cultural Activities	
VGVUCC203		National Service Scheme (NSS)	
VGVUCC204		Sports Activities	
VGVUCC205		Yoga	
		Total Credits	**22**

SEMESTER I

B. Sc (Information Technology)		Semester – I	
Course Name: Principles of Programming Languages using C		Course Code: VGVUSTMPP101	
Periods per week (1 Period is 60 minutes)		2	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2	60
	Internal	--	40

Course Objective
To make learners understand and use
1. Concept of programming language.
2. Concept of operators, data input and output, Conditional statements and loops in C.
3. Concept of Arrays and Functions in C.
4. Concept of structure in C.
5. Concept of pointers in C.

Unit	Details	Lectures
I	**Introduction :** Types of programming language, Introduction to C programming language, importance of C, sample C program, basic structure of C program, programming style, Compilation and execution of a C program **Fundamentals :** Character set, C tokens, Keywords and identifiers, Constants, Variables, Data types, Declaration of variables, assigning values to variables, Defining symbolic constants. **Operators and Expression :** Introduction, Arithmetic of Operators, Relational operators, Logical operators, Assignments operators, Increment and Decrement operators, Conditional operator, Bitwise operators, Special operators, Arithmetic expressions, Precedence of arithmetic operators, Type conversion in expression, operator precedence and associativity, Mathematical functions. **Managing Input and Output :** Reading a character, Writing a character, formatted input, formatted output	10

II	**Decision Making and branching :** Introduction, Simple if statement, The if else statement, nesting of if else statement, The else if ladder, The switch statement, The goto statement **Decision Making and looping :** Introduction, The while statement, The do while statement, The for statement, Jumps in loops **Arrays :** Introduction, One Dimensional arrays, Two Dimensional arrays, initializing Two Dimensional arrays, Multidimensional arrays. **User-Defined Functions :** Introduction, Need for user defined functions, Defining function, Declaring function, Calling function, Category of functions, Nesting of functions, Recursion, functions with arrays, Scope and lifetime of variables in functions.	10
III	**Structures :** Introduction, Structure definition, Declaring structure variables, Accessing structure members, Structure initialization, Arrays of structures, Arrays within structures. **Pointers :** Introduction, Declaring and initializing pointers, pointer expression, pointers and arrays.	10

Course Outcome
Learner will be able to
CO1 Understand various concepts of C language.
CO2 Understand the use of input, output statements of C.
CO3 Understand the concept and write the C language program using various looping statements and decision making statements., structures and pointers.
CO4 Understand the concept and write the C language program using arrays.
CO5 Understand the concept and write the C language program structures and pointers.

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Programming in ANSI	E. Balagurusamy	Tata McGRAW-Hill		
2.	Programming with C	Byron Gottfried	Tata McGRAW-Hill	2nd	1996
3	Programming Logic and Design	Joyce Farell	Cengage Learning	8th	2014
4	Let us C	Yashwant P. Kanetkar	BPB publication		
5	"C" Programming"	Brian W. Kernighan and Denis M. Ritchie.	PHI	2nd	

B. Sc (Information Technology)		Semester – I	
Course Name: Principles of Programming Languages using C Practical		Course Code: VGVUSTMP101	
Periods per week (1 Period is 120 minutes)		1	
Credits		1	
		Hours	Marks
Evaluation System	Practical Examination	2	50
	Internal	--	--

Course Objective
To make learners able to develop programs
1. Using variables, operators
2. To demonstrate use of decision making and looping statements using C.
3. To implement the concept of Arrays and Functions in C.
4. To demonstrate use of structure in C.
5. To demonstrate the concept of pointers in C.

List of Practical:	
1.	**Basic Programs:**
a.	Write a program to find the addition, subtraction, multiplication and division of two numbers.
b.	Write a program to swap two numbers without using a third variable.
2.	**Programs on variables:**
a.	Write a program to find the area of rectangle, square and circle.
b.	Write a program to find the volume of a cube, sphere, and cylinder.
c.	Write a program to find the largest of three numbers.
3.	**Conditional statements and loops(basic):**
a.	Write a program to enter a number from the user and display the month name. If number >13 then display invalid input using switch case.
b.	Write a program to check whether the number is even or odd.
c.	Write a program to check whether the number is positive, negative or zero.

d.	Write a program to find the factorial of a number.
e.	Write a program to check whether the entered number is prime or not.
4.	**Conditional statements and loops(advanced):**
a.	Write a program to find the sum of squares of digits of a number.
b.	Write a program to reverse the digits of an integer.
c.	Write a program to print the Fibbonacci series.
d.	Write a program to find whether a given number is palindrome or not.
e.	Write a program to count the digit in a number.
5.	**Functions:**
a.	Programs on Functions.
6.	**Recursive Functions:**
a.	Write a program to find the factorial of a number using a recursive function.
b.	Write a program to find the sum of natural numbers using a recursive function.
7.	**Arrays:**
a.	Write a program to find the largest value that is stored in the array.
b.	Write a program to arrange the 'n' numbers stored in the array in ascending and/or descending order.
c.	Write a program that performs addition, subtraction and multiplication of matrices.
8.	**Pointers and Structures:**
a.	Write a program to perform addition and subtraction of two pointer variables.
b.	Define a structure named Learner that stores the roll number, name, class, marks of five subjects. Write a C program that will take the Information of Learner from user and display marksheet.

Course Outcome
Learner will be able to
CO1 Write a simple program using C language.
CO2 Implement the C language Concepts like variables, operators.
CO3 Implement the C language program using arrays, structures and pointers.
CO4 Implement the C language program with user defined functions.
CO5 Motivate them to develop projects / applications using C language.

B. Sc. (Information Technology)		Semester – I	
Course Name: Microprocessor Architecture		Course Code: VGVUSTMMA101	
Periods per week (1 Period is 60 minutes)		2	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2	60
	Internal	--	40

Course Objective
To make learner understand
1. The microprocessor architecture and memory interface.
2. I/O interfacing with microprocessor and 8085 basic instructions.
3. The assembly programs using 8085.
4. The concept of looping, counters, and time delays in 8085 with instructions.
5. The concept of stack and subroutine.

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 operations, 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 the 8155 Memory Segment, Illustrative Example: Designing Memory for the MCTS Project, Testing and Troubleshooting Memory Interfacing Circuit,8085-Based SingleBoard microcomputer.	10
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.	10

	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.	
III	**Programming Techniques With Additional Instructions:** Programming Techniques: Looping, Counting and Indexing, Additional Data Transfer and 16-Bit Arithmetic Instructions Arithmetic Instruction Related to Memory, Logic Operations: Rotate, Logics Operations: Compare, Dynamic Debugging. **Counters and Time Delays:** Counters and Time Delays, Illustrative Program: Hexadecimal Counter, Illustrative Program: zero-to-nine (Modulo Ten) Counter, Generating Pulse Waveforms, Debugging Counter and Time-Delay Programs **Stacks and Sub-Routines:** Stack, Subroutine, Restart, Conditional Call, Return Instructions, Advanced Subroutine concepts.	**10**

Course Outcome
Learners should be able to
CO1 Understand basic knowledge of microprocessors.
CO2 Understand the architecture of a microprocessor 8085.
CO3 Map memory with processor.
CO4 Understand the instruction set of 8085.
CO5 Understand the concepts of timers, delays, counters, stacks, and subroutines in relation to the 8085.

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Microprocessors Architecture, Programming and Applications with the 8085.	Ramesh Gaonkar	PENRAM	Fifth	2012
2.	Computer System Architecture	M. Morris Mano	PHI		1998
3.	Structured Computer Organization	Andrew C. Tanenbaum	PHI		

B. Sc. (Information Technology)		Semester – I	
Course Name: Microprocessor Architecture Practical		**Course Code: VGVUSTMP101**	
Periods per week (1 Period is 120 minutes)		**1**	
Credits		**1**	
		Hours	**Marks**
Evaluation System	**Practical Examination**	**2**	**50**
	Internal	**--**	**--**

Course Objective
To make learner implement 8085-based programs to
1. Perform memory operations.
2. Carry out operations like addition, subtraction, 1's complement, 2's complement.
3. Perform Register based operations.
4. Carry out operations with multiple memory locations.
5. Perform operations with respect to memory locations.

List of Practical	
1.	**Perform the following Operations related to memory locations:**
a.	Exchange the contents of memory locations C200H and C201H.
b.	Add the contents of memory locations C200H and C201H and place the result in the memory locations C202H and C203H. Subtract the contents of memory location C201H from the memory location C200H and place the result in memory location C004H.
2.	**Simple assembly language programs I:**
a.	Add the 16-bit number in memory locations C200H and C201H to the 16-bit number in memory locations C202H and C203H. The most significant eight bits of the two numbers to be added are in memory locations C200H and C202H. Store the result in memory locations C204H and C205H with the most significant byte in memory location 4005H.

b.	Subtract the 16-bit number in memory locations C202H and C203H from the 16- bit number in memory locations C200H and C201H. The most significant eight bits of the two numbers are in memory locations C201H and C203H. Store the result in memory locations C204H and C205H with the most significant byte in memory location C205H.
3.	**Simple assembly language programs II :**
a.	Find the l's complement of the number stored at memory location C200H and store the complemented number at memory location C300H.
b.	Find the 2's complement of the number stored at memory location C200H and store the complemented number at memory location C300H.
4.	**Register Operations I:**
a.	Write a program to shift 8-bit data four bits right. Assume that data is in register C.
b.	Program to shift 16-bit data 1 bit left. Assume data is in the HL register pair
c.	Write a program to count the number of l's in the contents of D register and store the count in the B register.
5.	**Multiple memory locations I:**
a.	Multiply two 8-bit numbers stored in memory locations C200H and C201H by repetitive addition and store the result in memory locations C300H and C301H.
b.	Divide the 16-bit number stored in memory locations C200H and C201H by the 8 bit number stored at memory location C202H. Store the quotient in memory locations C300H and C301H and remainder in memory locations C302H and C303H.
6.	**Multiple memory locations II:**
a.	Find the number of negative elements (most significant bit 1) in a block of data. The length of the block is in memory location C200H and the block itself begins in memory location C201H. Store the number of negative elements in memory location C300H.
b.	Find the largest number in a block of data. The length of the block is in memory location C200H and the block itself starts from memory location C201H. Store the Maximum number in memory location C300H. Assume that the numbers in the block are all 8-bit unsigned binary numbers.
7.	**Calculations with respect to memory locations:**
a.	Write a program to sort given 10 numbers from memory location C200H in the ascending order.

b.	Calculate the sum of a series of even / odd numbers from the list of numbers. The length of the list is in memory location C200H and the series itself begins from memory location C201H. Assume the sum to be an 8 bit number so you can ignore carries and store the sum at memory location C250H.
8.	**Calculations with respect to memory locations:**
d.	Find the square of the given numbers from memory location C100H and store the result from memory location C200H.
e.	A list of 50 numbers is stored in memory, starting at C200H. Find the number of negative, zero and positive numbers from this list and store these results in memory locations C300H, C301H, and C302H respectively.

Course Outcome
Learners should be able
CO1 To write 8085 based simple assembly language programs.
CO2 To implement 8085 programs for handling memory.
CO3 To write a program for handling 8085 registers.
CO4 To carry out various arithmetic operations using 8085 instructions.
CO5 To write an 8085 program to carry out calculations with respect to memory.

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Microprocessors Architecture, Programming and Applications with the 8085.	Ramesh Gaonkar	PENRAM	Fifth	2012
2.	8080A/8085 Assembly Language Programming	Lance A. Leventhel	Osborne		1978

B. Sc. (Information Technology)		Semester – I	
Course Name: Discrete Mathematics		**Course Code: VGVUSTVSE101**	
Periods per week (1 Period is 60 minutes)		**2**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	**2**	**60**
	Internal	**--**	**40**

Course Objective
To make learner understand / implement
1. Concept of set theory
2. Elementary Number Theory and Methods of Proof.
3. Use of Addition Rule and probability concepts.
4. Concept of Relations, graphs and trees.
5. Sequences, Mathematical Induction, and Recursion.

Unit	Details	Lectures
I	**Introduction**: Variables, The Language of Sets, The Language of Relations and function. **Functions**: Functions Defined on General Sets, One-to-One and Onto, Inverse Functions, Composition of Functions, and Cardinality with Applications to Computability. **Relations**: Relations on Sets, Reflexivity, Symmetry, and Transitivity, Equivalence Relations, Partial Order Relations. **Set Theory**: Definitions and the Element Method of Proof, Properties of Sets, Disproof's, Algebraic Proofs, Boolean Algebras, Russell's Paradox and the Halting Problem.	**10**
II	**The Logic of Compound Statements**: Logical form and Logical Equivalence, Conditional Statements, Valid and Invalid Arguments. **Quantified Statements**: Predicates and Quantified Statements, Statements with Multiple Quantifiers, Arguments with Quantified Statements. **Elementary Number Theory and Methods of Proof**: Introduction to Direct	**10**

	Proofs, Rational Numbers, Divisibility, Division into Cases and the QuotientRemainder Theorem, floor and Ceiling, Indirect Argument Contradiction and Contraposition, Two Classical Theorems, Applications in algorithms. **Probability** : Basics of Probability, Addition rule	
III	**Sequences, Mathematical Induction, and Recursion**: Sequences, Mathematical Induction, Strong Mathematical Induction and the Well-Ordering Principle or the Integers, Correctness of algorithms, defining sequences recursively, solving recurrence relations by iteration, Second order linear homogeneous recurrence relations with constant coefficients. General recursive definitions and structural induction. **Graphs and Trees**: Definitions and Basic Properties, Trails, Paths, and Circuits, Matrix Representations of Graphs, Isomorphism of Graphs, Trees, Rooted Trees, Isomorphism of Graphs, Spanning trees and shortest paths	**10**

Course Outcome
Learners should be able to
CO1 Understand the basic principles of sets and operations in sets.
CO2 Understand the working with relations and investigate their properties.
CO3 Analyse mathematical properties using mathematical induction methods, study functions, spaces, and other mathematical structures using sequences and use of recursion.
CO4 Understand relation, graphs and trees in various applications.
CO5 Understand the use of the SCILAB tool to solve mathematical problems.

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Discrete Mathematics with Applications	Sussana S. Epp	Cengage Learning	4th	2010
2.	Discrete Mathematics, Schaum's Outlines Series	Seymour Lipschutz, Marc Lipson	Tata MCGraw Hill		2007
3.	Discrete Mathematics and its Applications	Kenneth H. Rosen	Tata MCGraw Hill		
4.	Discrete mathematical structures	B KolmanRC Busby, SRoss	PHI		
5.	Discrete structures	Liu	Tata MCGraw Hill		

B. Sc. (Information Technology)		Semester – I	
Course Name: Numerical Computation using Scilab Practical		**Course Code: VGVUSTVSEP101**	
Periods per week (1 Period is 120 minutes)		**2**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Practical Examination**	3	**100**
	Internal	**--**	**--**

Course Objective
To make learner understand / implement
1. To be able execute programs of discrete mathematics concepts in Scilab software easily.
2. To implement programs on Set theory, functions and algorithms.
3. To execute different topics related to probability using the Scilab software.
4. To represent concepts of graph theory, directed graphs, and their subtopics in the form of a program.
5. To implement topics like Algebraic systems, Boolean algebra, Recurrence relations in a practical manner.

List of Practical: Write the programs or the following using SCILAB	
1.	**Set Theory :**
a.	Inclusion Exclusion principle.
b.	Power Sets
2.	**Functions and Algorithms :**
a.	Recursively defined Functions
b.	Cardinality
c.	Polynomial evaluation
d.	Greatest Common Divisor
3.	**Sequences :**
a.	Summation Notation, Product Notation

b.	Mathematical Induction
4.	**Probability Theory :**
a.	Sample space and events
b.	Finite probability spaces
5.	**Counting Elements Of Disjoints Sets :**
a.	Addition Principle
b.	Conditional Probability
c.	Independent events
d.	Repeated trials with two outcomes
6.	**Directed Graphs :**
a.	Adjacency matrix
b.	Path matrix
7.	**Undirected Graphs :**
a.	Adjacency matrix
b.	Path matrix
8.	**Graph Theory :**
a.	Paths and connectivity
b.	Minimum spanning tree
c.	Isomorphism
9.	**Tree :**
a.	Minimum spanning tree
b.	Shortest path algorithm Kruskal or Prims
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 Outcome
To make learner understand / implement
1. Learner will be able to become familiar with the Scilab environment
2. Implement programs on Inclusion Exclusion principle, power sets, recursively defined functions, Mathematical Induction Cardinality in Scilab
3. Execute programs like Sum principle, Product principle, Factorial, Permutations and Combinations, Sample space and events, Conditional Probability, Finite probability spaces
4. Implement concepts in Scilab like paths and connectivity, minimum spanning tree, isomorphism, adjacency matrix, path matrix.
5. Implement recurrence relations by iteration, Second order linear homogenous recurrence relations with constant coefficients

B. Sc (Information Technology)		Semester – I	
Course Name: Environmental Study for Sustainable IT I		**Course Code: VGVUVE108**	
Periods per week (1 Period is 60 minutes)		**2**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	**2**	**60**
	Internal		**40**

Course Objective
To aid learner to
1. Understand the concept of Green IT , green IT's hardware and software aspects,
2. Understand how software characteristics impact the sustainability or greenness of computing applications, and outlines the notion of sustainable software engineering.
3. Understand Key sustainability challenges associated with data centers and strategies to minimize data center's energy consumption and carbon footprint.
4. Understand regulatory, non regulatory and other influences affecting business and the IT industry to make them more environmentally sustainable.
5. Get in-depth coverage of energy-efficient storage technologies and data storage systems.

Unit	Details	Lectures
I	**Green IT An Overview :** Introduction, Environmental Concerns and Sustainable Development, Environmental Impacts of IT, Green IT, Holistic Approach to Greening IT, Greening IT, Applying IT for Enhancing Environmental Sustainability, Green IT Standards and Eco-Labelling of IT. **Green Devices and Hardware :** Introduction, Life Cycle of a Device or Hardware, Reuse, Recycle and Dispose **Green Software :** Introduction, Energy-Saving Software Techniques, Evaluating and Measuring Software Impact to Platform Power.	10

II	**Sustainable Software Development :** Introduction, Current Practices, Sustainable Software, Software Sustainability Attributes, Software Sustainability Metrics,Sustainable Software Methodology, Defining Actions. **Regulating Green IT: Laws, Standards and Protocols :** Introduction, Introduction, Nonregulatory Government Initiatives, Industry Associations and Standards Bodies, Green Building Standards, Green Data Centres, Social Movements and Greenpeace.	10
III	**Green Data Storage :** Introduction, Storage Media Power Characteristics, Energy Management Techniques for Hard Disks, System-Level Energy Management. **Green Data Centres :** Data Centres and Associated Energy Challenges, Data Centre IT Infrastructure, Data Centre Facility Infrastructure: Implications for Energy Efficiency, IT Infrastructure Management, Green Data Centre Metrics	10

Course Outcome
Learners should be able to
CO1 Discusses the scope of emerging green IT regulations and public policy.
CO2 Identify the energy management techniques.
CO3 Know laws, standards and regulations related to Green IT.
CO4 Discuss how the choice of hardware and software can facilitate a more sustainable operation.
CO5 Develop knowledge about green data storage and data centers.

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. Gangadharan	WILEY		-
3.	Green Data Center: Steps for the Journey	Alvin Galea, Michael Schaefer,Mike Ebbers	Shroff Publishers And Distributors		2011
4.	Green Computing and Green IT Best Practice	Jason Harris	Emereo		
5.	Green Computing Tools and Techniques for Saving Energy, Money and Resources	Bud E. Smith	CRC Press		2014

B. Sc (Information Technology)		Semester – I	
Course Name: Ancient Vedic Mathematics		Course Code: VGVUIKS106	
Periods per week (1 Period is 60 minutes)		2	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	02	60
	Internal		40

Course Objective
To make learner
1. Enhance computation skills in students through Vedic Mathematics
2. Develop logical and analytical thinking
3. Promote joyful learning of mathematics
4. Discuss the rich heritage of mathematical temper of Ancient India.
5. Foster love for math's and remove its fear through Vedic Mathematics

Unit	Details	Lectures
I	**High Speed Addition, Subtraction, Miracle Multiplication :** **Vedic Maths**: History of Vedic Maths and its Features **Vedic Maths formulae:** Sutras and Upsutras **Addition in Vedic Maths:** Without carrying, Dot Method **Subtraction in Vedic Maths:** Nikhilam Navatashcaramam Dashatah Fraction-Addition and Subtraction **Multiplication in Vedic Maths:** Base Method (any two numbers upto three digits), Multiplication by Urdhva Tiryak Sutra **Miracle multiplication:** Any three-digit number by series of 1's and 9's.	10
II	**Excellent Division , Lightening Squares :** Division by Urdhva Tiryak Sutra (Vinculum method) , Squares of any two-digit numbers: Base method, Square of numbers ending in 5: Ekadhikena Purvena Sutra,Easy square roots: Dwandwa Yoga (duplex) Sutra, Square root of 2: Baudhayana Shulbasutra	10

III	**Rapid Cubes, Enlighten Algebra and Geometry :** Cubing: Yavadunam Sutra **FactoringQuadratic equation:** Anurupyena, Adyamadyenantyamantya Sutra Concept of Baudhayana (Pythagoras) Theorem **Circling a square:** Baudhayana Shulbasutra **Concept of pi:** Baudhayana Shulbasutra. **Concept angle (8) 0°, 30°, 45°, 60° and 90°:** Baudhayana number	**10**

Course Outcome
Learners should be able to
CO1 Understanding the principles and techniques of Vedic mathematics as described in ancient Indian texts.
CO2 Developing mental calculation skills to perform arithmetic operations quickly and accurately.
CO3 Applying Vedic mathematics techniques to solve complex mathematical problems efficiently.
CO4 Improving problem-solving abilities and logical reasoning through the study of Vedic mathematics.
CO5 Enhancing concentration, memory, and cognitive abilities through practicing Vedic mathematics techniques.

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	The Essential of Vedic Mathematics	Rajesh Kumar Thakur	Rupa Publications	-	2019
2.	Vedic Mathematics Made Easy.	Dahaval Bathia	Jaico Publishing,	-	2011
3	Vedic Mathematics: Sixteen Simple Mathematical formulae from the VedasMotilal Banarasidas,	Jagadguru Swami Sri Bharati Krishna Trithaji,	-	-	2015.
4	Learn Vedic Speed Mathematics Systematically	Chaitnaya A. Patil	-	-	2018.
5	A Modern Introduction to Ancient Indian Mathematics	TS Bhanumurth	Wiley Eastern Limited	-	-

SEMESTER II

B. Sc. (Information Technology)		Semester – II	
Course Name: Object Oriented Programming using C++		**Course Code: VGVUSTMOP201**	
Periods per week (1 Period is 60 minutes)		**2**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	**2**	**60**
	Internal	**--**	**40**

Course Objective
To make learner understand / implement
1. Basic Object Oriented Methodology and principles.
2. Concept of classes and objects, constructors and destructors.
3. Polymorphism and virtual functions.
4. Inheritance and exception handling.
5. File handling and OOPs concepts using C++.

Unit	Details	Lectures
I	**Object Oriented Methodology**: Introduction, Advantages and Disadvantages of Procedure Oriented Languages, Application of OOPS, Principles of OOPS: Objects, Classes, Data Abstraction and Data Encapsulation, Inheritance, Polymorphism, Dynamic Binding, Message Passing. **Classes and Objects:** Simple classes (Class specification, class members accessing), Defining member functions, passing object as an argument, Returning object from functions, friend classes, friend function. **Constructors and Destructors:** Introduction, Default Constructor, Parameterized Constructor and examples, Destructors.	**10**

II	**Program development using Inheritance:** Introduction, Advantages provided by inheritance, choosing the access specifier, Derived class declaration, derived class constructors, class hierarchies, multiple inheritance, multilevel inheritance, hybrid inheritance. **Polymorphism:** Concept of function overloading, overloaded operators, overloading unary and binary operators.	**10**
III	**Virtual Functions:** Introduction and need, Pure Virtual Functions, this Pointer, abstract classes, virtual destructors. **Exception Handling:** Introduction, Exception Handling Mechanism, Concept of throw & catch with example. **Working with Files:** Introduction, File Operations, Various File Modes, File Pointer and their Manipulation.	**10**

Course Outcome
Learners should be able use C++ language to
CO1 Implement the basic principles of OOPs
CO2 Implement Object Oriented Concepts like classes , constructors, destructors etc.
CO3 Implement various types of inheritance, polymorphism and virtual function.
CO4 Implement exception and file handling.
CO5 Develop a project / application using C++.

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Object Oriented Analysis and Design	Timothy Budd	TMH	3rd	2012
2.	Mastering C++	K R Venugopal, Rajkumar Buyya, T Ravishankar	Tata McGraw Hill	2nd	2011
3.	C++ for beginners	B. M. Hirwani	SPD		2013
4.	Effective Modern C++	Scott Meyers	SPD		
5.	Object Oriented Programming with C++	E. Balagurusamy	Tata McGraw Hill	4th	
6.	Learning Python	Mark Lutz	O' Reilly	5th	2013
7.	Mastering Object Oriented Python	Steven F. Lott	Pact Publishing		2014

B. Sc. (Information Technology)		Semester – II	
Course Name: Object Oriented Programming using C++ Practical		**Course Code: VGVUSTMP201**	
Periods per week (1 Period is 120 minutes)		**1**	
Credits		**1**	
		Hours	**Marks**
Evaluation System	**Practical Examination**	**2**	**50**
	Internal	**--**	**--**

Course Objective
To make learner understand / implement
1. The installation process of software used for C++ and OOPs concepts.
2. The familiar of software and working of C++ programs.
3. Implementation of Libraries in programs.
4. Implementation of Object-Oriented Principles in C++ programs.
5. Implement various OOPs concepts using C++.

List of Practical:	
1.	**Classes and methods :**
a.	Design an employee class for reading and displaying the employee information, the getInfo() and displayInfo() methods will be used respectively. Where getInfo() will be private method
b.	Design the class Learner containing getData() and displayData() as two of its methods which will be used for reading and displaying the Learner information respectively. Where getData() will be a private method.
c.	Design the class Demo which will contain the following methods: readNo(), factorial() for calculating the factorial of a number, reverseNo() will reverse the given number, isPalindrome() will check the given number is palindrome, isArmstrong() which will calculate whether the given number is armStrong or not.WherereadNo() will be a private method.
d.	Write a program to demonstrate function definition outside class and accessing class members in function definition.
2.	**Using friend functions :**
a.	Write a friend function for adding the two complex numbers, using a single class

b.	Write a friend function for adding two different distances and display its sum, using two classes.
c.	Write a friend function for adding two matrices from two different classes and display its sum.
3.	**Constructors and method overloading :**
a.	Design a class Complex for adding the two complex numbers and also show the use of constructor.
b.	Design a class Geometry containing the methods area() and volume() andalso overload the area() function .
c.	Design a class StaticDemo to show the implementation of static variables and static functions.
4.	**Operator Overloading :**
a.	Overload the operator unary(-) for demonstrating operator overloading.
b.	Overload the operator+ for adding the timings of two clocks, And also pass objects as an argument.
c.	Overload the + for concatenating the two strings. For e.g "Py" + "thon" = Python
5.	**Inheritance :**
a.	Design a class for single level inheritance using public and private type derivation.
b.	Design a class for multiple inheritance.
c.	Implement hierarchical inheritance.
6.	**Virtual functions and abstract classes :**
a.	Implement the concept of method overriding.
b.	Show the use of virtual function
c.	Show the implementation of abstract class.
7.	**Exception handling :**
a.	Show the implementation of exception handling
b.	Show the implementation for exception handling for strings
c.	Show the implementation of exception handling for using the pointers.
8.	**File handling :**
a.	Design a class FileDemo opens a file in read mode and displays the total number of words and lines in the file.
b.	Design a class to handle multiple files and file operations
c.	Design a editor for appending and editing the files

Course Outcome
Learners should be able use C++ language to
CO1 Implement the syntax and semantics of C++ programming.
CO2 Understand how to model real world scenarios using class and be able to exhibit the communication between objects.
CO3 Understand the reusability of programs using the concepts of inheritance and polymorphism.
CO4 Apply the concepts of Exception handling to develop efficient and error free codes.
CO5 Implement basic programs which help to read and write the data in a file using file handling.

B. Sc. (Information Technology)		Semester – II	
Course Name: Database Management System		**Course Code: VGVUSTMDS201**	
Periods per week (1 Period is 60 minutes)		**2**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	**2**	**60**
	Internal	**--**	**40**

Course Objective
To make learner
1. Aware about the concept of database management system with the help of various models of database design. Will be able to design the database schema with the use of ER diagram.
2. Concept of operators, data input and output, Conditional statements and loops in C.
3. Aware of using different constraints for database creation, writing various SQL queries and views concept as well as implement the database concepts using Oracle 11g.
4. Will study transaction management in DBMS.
5. Understand and implement Pl/SQL concept using Oracle 11g.

Unit	Details	Lectures
I	**Introduction :** Database System Applications, Database Systems versus File Systems, View of data, Database languages, Database users and administrators, Transaction management, database architecture. **Data Models** **Entity-relationship model** – Basic concepts. Constraints, Keys, Design Issues, Entity-relationship Diagram, Weak Entity Sets, Extended E-R Features, Reduction of an E-R Schema to Tables. **Relational Model** - Structure of Relational Databases, Basic Structure, Database Schema, Keys, Schema Diagram, Query Languages.	10

II	**SQL :** Introduction, The Role of SQL,SQL Features and Benefits **Database Structure :** The Data Definition Language, Table Definitions, Constraint Definition **Updating Data :** Adding Data to Database, Deleting data from Database, Modifying data in database. **Retrieving Data :** SQL Basics, Simple Queries, Multitable Queries, Summary Queries, Subqueries. **Views:** What is view?, Creating View, Updating View, Dropping View. **Transaction :** What is a transaction? Transaction Concept, ACID Properties, Transaction State.	10
III	**PL-SQL**: Introduction,Generic PL/SQL block, Execution Environment, The character set, literals, Data types, Variables, Constants, Logical C\comparisons, Disp[laying user messages on the screen, Comments, Conditional control,Iterative control. **Cursor :** Introduction, Types of cursor, Implicit Cursor, Explicit Cursor, Cursor For Loop, Parameterized cursor. **Error Handling :** Error handling in PL/SQL, Oracle's named Exception Handlers, User Defined Exception Handler for I/O validation and Business Rule validation. **Database Objects :** Stored Procedures and functions, Advantages, Procedures versus Functions, Creating stored Procedure and Functions, Deleting Stored Procedure and Functions, Database Triggers - Introduction, use of database trigger, Database Trigger versus Procedures, Database Trigger versus Declarative Integrity Constraints, Applying Database /triggers, Types of Triggers, Declaring a Trigger.	10

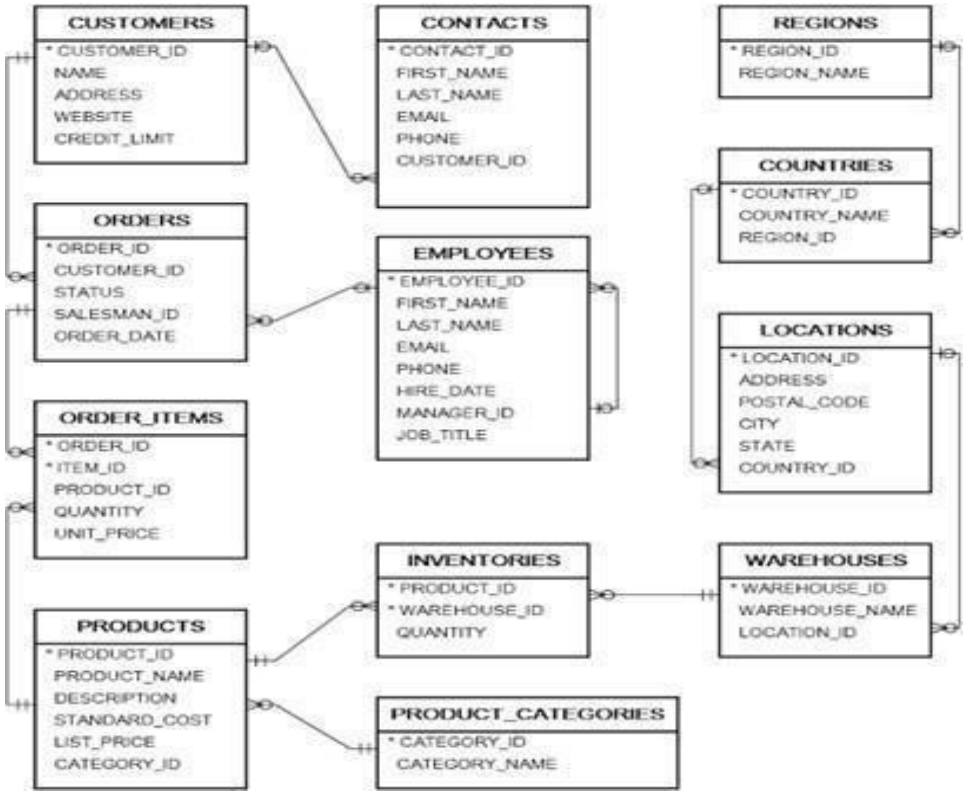
Course Outcome
Learners should be able to
CO1 Understand the importance of database management system.
CO2 Create database schema using ER model and Relational Model.
CO3 Write various SQL queries, apply various constraints, creating database objects.
CO4 Apply PL SQL concepts.
CO5 Motivate the student to work as database administrator.

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Database System and Concepts	A Silberschatz, H Korth, S udarshan	McGraw-Hill	Fifth Edition	
2.	Database Systems	RobCoronel	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 – II	
Course Name: Database Management System Practical		Course Code: VGVUSTMP201	
Periods per week (1 Period is 120 minutes)		1	
Credits		1	
		Hours	Marks
Evaluation System	Practical Examination	2	50
	Internal	--	--

Course Objective
To make learner
1. Aware of various DDL statements used for database creation.
2. Aware of various DDL statements used for modifying data in the database.
3. Aware of various SQL statements used to retrieve data from the database.
4. Aware of creating and using concept of views.
5. Understand PL/SQL concepts.

Consider the following schema for practices.



List of Practical	
1.	**Creating and Managing Tables :**
a.	Creating Tables for the above schema with constraints defined (if required).
b.	Insert at least 5 rows in each table.
2.	**SQL Statements – 1 :**
a.	Writing Basic SQL SELECT Statements
b.	Display the rows of table in Sorted order (ascending / descending)
c.	Use various single row function like Upper, Lower, Concat, Length, SUBSTR,TRIM, LTRIM, RTRIM REPLACE
3.	**SQL Statements – 2 :**
a.	Displaying Data from Multiple Tables
b.	Subqueries
c.	Use various aggregrate functions like min, max, sum, avg, count
4.	**Manipulating Data :**
a.	Use Update statement to update the rows of tables.
b.	Illustrate use of DELETE statement
5.	**Creating and Managing other database objects :**
a.	Creating Views
b.	Other Database Objects like sequence, index and synonym
c.	Controlling User Access
6.	**Using SET operators, Date/Time Functions, GROUP BY clause (advanced features) and advanced subqueries :**
a.	Using SET Operators
b.	Use various Date Functions like NEXT_DAY, LAST_DAY, TO_CHAR,ADD_MONTHS, CURRENT_DATE, SYSDATE
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 , Functions,
b.	Managing Subprograms

Course Outcome
Learners should be able
CO1 To create and alter the database structure.
CO2 To fire SQL queries based on INSERT, UPDATE, SELECT and DELETE statements.
CO3 To execute queries based on date, strings.
CO4 To create and manage views.
CO5 To write a PL/SQL code block based on cursor, procedure, function and, exception.

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	SQL The Complete Reference	Paul Weinberg, James Groff, Andrew Oppel	McGraw-Hill	Third	
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 – II	
Course Name: Basics of Data Science		Course Code: VGVUSTNDS201	
Periods per week (1 Period is 60 minutes)		2	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2	60
	Internal	--	40

Course Objective
At the end of the course, a learner will be able to
1. Describe the significance of data science.
2. Understand the Data Science process.
3. Explain how data is collected, managed and stored for data science.
4. Build, and prepare data for use with a variety of statistical methods and models.
5. Analyze Data using various Visualization techniques.

Unit	Details	Lectures
I	**Introduction:** What Is Data Science?, Importance and Need of Data science, Data Science Process , Prerequisite for Data Scientist, Components of Data Science, The Industry Applications Of Data Science Real-Life Examples Of Data Science, Data Science Terminologies, Computer Science, Data Science, and Real Science, Properties of Data, Classification and Regression **Mathematical Preliminaries:** Probability, Descriptive Statistics, Correlation Analysis, Logarithms	10
II	**Data Munging :** Languages for Data Science, Collecting Data, Cleaning Data, Crowdsourcing **Scores and Rankings:** The Body Mass Index(BMI), Developing Scoring Systems, Z-scores and Normalization,	10
III	**Statistical Analysis:** Statistical Distributions, Sampling from Distributions, Statistical Significance, Permutation Tests and P-values **Visualizing Data:** Exploratory Data Analysis, Developing a Visualization Aesthetic, Chart Types, Great Visualizations, Reading Graphs	10

Course Outcome
Learners will be able to
CO1 Understand the basics of data science.
CO2 Understand the concepts of Data collection and management.
CO3 Establish sources of data and various mathematical concepts for Data Science.
CO4 Use the concepts of statistics and Identify distribution properties of data using statistical concepts.
CO5 Understand types of data Visualization techniques.

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	The Data Science Design Manual	Steven S. Skiena	Springer	1st	2017
2.	Foundations of Data Science	Avrim Blum, John Hopcroft, and Ravindran Kannan			
3.	Doing Data Science	Rachel Schutt and Cathy O'Neil	O'Reilly	First Edition	2013
4.	Data Science from Scratch	Joel Grus	O'Reilly	Second Edition	2019
5.	Fundamentals of Data Science	Sanjeev J. Wagh, Manisha S. Bhende, Anuradha D. Thakare	Chapman & Hall	First Edition	2021

B. Sc (Information Technology)		Semester – II	
Course Name: Fundamentals of Digital Electronics		**Course Code: VGVUSTVSE201**	
Periods per week (1 Period is 60 minutes)		**2**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	**2**	**60**
	Internal	**--**	**40**

Course Objective
To make learner understand concept of
1. Number system and binary arithmetic.
2. Boolean algebra, logic gates and Karnaugh Map.
3. Combinational Logic circuit and arithmetic circuits.
4. Multiplexer, demultiplexer, encoder, decoder and flip-flops.
5. Counters and shift registers

Unit	Details	Lectures
I	**Number System:** Analog System, digital system, Numbering system(Binary, Octal, Hexadecimal), conversion from one number system to another, weighted codes, Error detection and correction, Code Conversion. **Binary Arithmetic:** Binary addition, Binary subtraction, multiplication and division, Negative number representation, Subtraction using 1's complement and 2's complement. **Logic Gates:** Introduction to Logic gates, Exclusive OR and Exclusive NOR gates, Universal Logic gates, Implementation of other gates using universal gates.	**10**
II	**Boolean Algebra:** Boolean theorems and Laws, De Morgan's Theorem, Perfect Induction, Reduction of Logic expression using Boolean Algebra, Deriving Boolean expression from given circuit, **Minterm, Maxterm, and Karnaugh Maps:** Introduction, minterms and sum of minterm form, maxterm and Product of maxterm form, Reduction technique	**10**

	using Karnaugh maps – 2/3/4 variable K-maps, Grouping of variables in Kmaps, minimize Boolean expression using K- map.	
III	**Combinational Logic Circuits:** Introduction, Multi-input, multi-output Combinational circuits, Code converters design and implementations. **Arithmetic Circuits:** Introduction, Adders, Subtractor, Multiplier, Comparator. **Multiplexer, Demultiplexer, Encoder and Decoder:** Introduction, Multiplexer, Demultiplexer, Decoder, Encoders. **Sequential Circuits:** Introduction to flip-flop and its types.	**10**

Course Outcome
Learners should be able to
CO1 Perform conversions among different number systems, become familiar with basic logic gates and understand Boolean algebra.
CO2 Understand the use of K-map for hardware minimization.
CO3 Understand the design of combinational circuits such as multiplexer, demultiplexer, encoder and decoder etc.
CO4 Understand the design of sequential Circuits such as flip-flops, Registers, and Counters.
CO5 Obtain a basic level of Digital Electronics knowledge and set the stage to perform the analysis and design of Complex Digital electronic Circuits.

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Digital Electronics and Logic Design	N. G. Palan	Technova		
2.	Make Electronics	Charles Platt	O'Reilly	1st	2010
3.	Modern Digital Electronics	R. P. Jain	Tata McGraw Hill	3rd	
4.	Digital Principles and Applications	Malvino and Leach	Tata McGraw Hill		
5.	Digital Electronics: Principles, Devices and Applications,	Anil K. Maini	Wiley		2007

B. Sc (Information Technology)		Semester – II	
Course Name: Fundamentals of Digital Electronics Practical		**Course Code: VGVUSTVSEP201**	
Periods per week (1 Period is 120 minutes)		**2**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Practical Examination**	**2**	**50**
	Internal	**--**	**--**

Course Objective
To make learner
1. Aware of electronic trainer kit and its Usage
2. How to use various integrated circuits for basic logic gates
3. To implement the applications of basic logic gates in Combinational circuits
4. To implement the applications of basic logic gates in Sequential circuits
5. To introduce and use Logisim simulator

List of Practical	
1.	**Study of Logic gates and their ICs and universal gates:**
a.	Study of AND, OR, NOT, XOR, XNOR, NAND and NOR gates.
b.	IC 7400, 7402, 7404, 7408, 7432, 7486, 74266.
2.	**Study of Universal Gates:**
a.	Implement AND, OR, NOT, XOR, XNOR using NAND gates.
b.	Implement AND, OR, NOT, XOR, XNOR using NOR gates.
3.	**Implement the given Boolean expressions using minimum number of gates:**
a.	Verifying De Morgan's laws.
b.	Implement other given expressions using a minimum number of gates.
c.	Implement other given expressions using a minimum number of ICs.
4.	**Implement combinational circuits:**

a.	Design and implement combinational circuit based on the problem given and minimizing using K-maps.
5.	**Implement code converters:**
a.	Design and implement Binary – to – Gray code converter.
b.	Design and implement Gray – to – Binary code converter.
c.	Design and implement Binary – to – BCD code converter
d.	Design and implement Binary – to – XS-3 code converter
6.	**Implement Adder and Subtractor Arithmetic circuits.:**
a.	Design and implement Half adder and Full adder.
b.	Design and implement Half subtractor and Full subtractor.
7.	**Implement Arithmetic circuits:**
a.	Design and implement a 2-bit by 2-bit multiplier.
b.	Design and implement a 2-bit comparator.
8.	**Implement Encode and Decoder:**
a.	Design and implement 8:3 encoder.
b.	Design and implement 3:8 decoder.
9.	**Implement Multiplexer and Demultiplexers:**
a.	Design and implement 4:1 multiplexer. Study of IC 74153, 74157
b.	Design and implement 1:4 demultiplexer. Study of IC 74139
10.	**Study of flip-flops :**
a.	Study of IC 7473.
b.	Study of IC 7474.
c.	Study of IC 7476.

Course Outcome
To make learner
1.To be familiar with the Electronic trainer kit.
2. To understand and design and implement combinational circuits.
3. To understand design and implement Sequential circuits.
4. To Perform simulation of various circuits on Logisim simulator
5. To develop applications (Half adder, Full adder) of combinational circuit and Sequential circuit(Flip Flop).

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Digital Electronics and Logic Design	N. G. Palan	Technova		
2.	Digital Principles and Applications	Malvino and Leach	Tata McGraw Hill		

B. Sc (Information Technology)		Semester – I	
Course Name: Effective Communication Skills		**Course Code:** VGVUFAE204	
Periods per week (1 Period is 60 minutes)		**2**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	**2**	**60**
	Internal		**40**

Course Objective
To make learner understand
1. Recognized seven C's of communication.
2. Importance of nonverbal communication.
3. Oral as well as written communication skills.
4. The importance of voice tone and body language in effective communication.
5. Communicate the message in businesses in an effective and engaging way.

Unit	Details	Lectures
I	**The Seven Cs of Effective Communication:** Completeness, Conciseness, Consideration, Concreteness, Clarity, Courtesy, Correctness. **Understanding Business Communication:** Nature and Scope of Communication, Non-verbal Communication, Technology-enabled Business Communication.	**10**
II	**Writing Business Messages and Documents:** Business writing, Business Correspondence, Instructions Business Reports and Proposals, Career building and Resume writing. **Developing Oral Communication Skills or Business:** Effective Listening, Business Presentations and Public Speaking, Conversations, Interviews	**10**

III	**Developing Oral Communication Skills or Business:** Meetings. Group Discussions and Team Presentations and team Briefing. **Understanding Specific Communication Needs:** Communication across functional Areas, Corporate Communication, Persuasive Strategies in Business Communication, Ethics in Business Communication.	**10**

Course Outcome
Learners should be able to
CO1 Understand and apply communication theory.
CO2 Develop and enhance understanding, and create reports with others.
CO3 Participate effectively in small group interactions.
CO4 Prepare and deliver effective oral presentations.
CO5 Improve and broaden your writing abilities for business messages and documents.

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Business Communication	Edited by Meenakshi Raman and Prakash Singh	Oxford University Press	Second	
2.	Professional Communication	Aruna Koneru	Tata McGraw Hill		
3.	Strategies for improving your business communication	Pro. M. S. Rao	Shroff publishers and distributors		2016
4.	Business Communication	Dr. Rishipal and Dr. Jyoti Sheoran	SPD		2014

B. Sc (Information Technology)		Semester – II	
Course Name: Environmental Study for Sustainable IT II		**Course Code: VGVUVE206**	
Periods per week (1 Period is 60 minutes)		**2**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	**2**	**60**
	Internal		**40**

Course Objective
To aid learner to
1. Understand issues of Green Computing as well as different standards.
2. Understand the modern approaches to Green Computing.
3. Understand the alternatives for cooling your data center.
4. Understand the need for making computer networks and communications energy efficient
5. Understand cloud computing in the context of environmental sustainability and various elements of clouds that contribute to total energy consumption

Unit	Details	Lectures
I	**Overview and Issues:** Problems: Toxins, Power Consumption, Equipment Disposal, Company's Carbon Footprint: Measuring, Details, reasons to bother, Plan for the Future, Cost Savings: Hardware, Power. **Initiatives and Standards:** Global Initiatives: United Nations, Basel Action Network, Basel Convention, North America: The United States, Canada, Australia, Europe, WEEE Directive, RoHS, National Adoption, Asia: Japan, China, Korea.	10
II	**Minimizing Power Usage:** Power Problems, Monitoring Power Usage, Servers, Low-Cost Options, Reducing Power Use, Data De-Duplication, Virtualization, Management, Bigger Drives, Involving the Utility Company, Low-Power Computers, PCs, Linux, Components, Servers, Computer Settings, Storage, Monitors, Power Supplies, Wireless Devices, Software.	10

	Cooling: Cooling Costs, Power Cost, Causes of Cost, Calculating Cooling Needs, 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**.**	
III	**Green Networks and Communications :** Introduction, Objectives of Green Network Protocols, Green Network Protocols and Standards. **Green Cloud Computing and Environmental Sustainability :** Introduction, What is Cloud Computing?, Cloud Computing and Energy Usage Model: A Typical Example, Features of Clouds Enabling Green Computing, Green Cloud Architecture	10

Course Outcome
Learners should be able to
CO1 Give an account of the concept green IT, environmental perspectives on IT use, standards and certifications related to sustainable IT products.
CO2 Describe green IT in relation to technology.
CO3 Evaluate IT use in relation to environmental perspectives.
CO4 Formulate plans for reducing IT heating and cooling requirements.
CO5 Implement Green IT in Real Life.

Books and References:

Sr. No.	Title	Author/s	Publisher	Edit ion	Year
1.	Green IT	Toby Velte, Anthony Velte, & Robert Elsenpete	McGraw Hill		2008
2.	Harnessing Green It Principles And Practices	San Murugesan, G.R. Gangadharan	WILEY		-
3.	Green Data Center: Steps for the Journey	Alvin Galea, Michael Schaefer, Mike Ebbers	Shroff Publishers And Distributors		2011
4.	Green Computing and Green IT Best Practice	Jason Harris	Emereo		
5.	Green Computing Tools and Techniques for Saving Energy, Money and Resources	Bud E. Smith	CRC Press		2014

Evaluation Scheme

(Major, Minor, VSEC)

1. Internal Evaluation (40 marks).

i. Test: 1 Class test of 15 marks. (Can be taken online)

Q	Attempt *any three* of the following:	15
a.		
b.		
c.		
d.		
e.		
f.		

ii. 15 marks project/presentation/assignment.

iii. 10 marks: Active participation in the class, overall conduct, attendance.

2. External Examination: (60marks)

All questions are compulsory		
Q.1.	(Based on Unit 1) Attempt *any three* of the following:	15
a.		
b.		
c.		
d.		
e.		
f.		
Q.2.	(Based on Unit 2) Attempt *any three* of the following:	15
Q.3.	(Based on Unit 3) Attempt *any three* of the following:	15
Q.4.	(Based on whole syllabus) Attempt *any three* of the following:	15

Practical Exam: 100marks

A Certified copy journal is essential to appear for the practical examination.

1.	Practical Question 1	**40**
2.	Practical Question 2	**40**
3.	Journal	10
4.	Viva Voce	10

OR

1.	Practical Question	**80**
2.	Journal	10
3.	Viva Voce	10

Evaluation Scheme

(AEC, VEC, IKS)

1Internal Evaluation (50 marks).

i. Test: 1 Class test of 15 marks. (Can be taken online)

Q	Attempt *any three of* the following:	15
a.		
b.		
c.		
d.		
e.		
f.		

ii. 25 marks project/presentation/assignment.

iii. 10 marks: Active participation in the class, overall conduct, attendance.

Evaluation Scheme

(CC)

1Internal Evaluation (50 marks).

i. Test: 1 Class test of 15 marks. (Can be taken online)

Q	Attempt ***any three*** ***of*** **the following:**	15
a.		
b.		
c.		
d.		
e.		
f.		

ii. 35 marks Activities

KET's V. G. Vaze College(Autonomous) Board of Studies – Information Technology comprised of following members

Mrs. Pournima P Bhangale, Chairperson

Mrs. Vandana Y Kadam, Member

Mrs. Rakhee D Rane, Member

Mrs. Nanda A Rupnar, Member

Ms. Mohini Bhole, Member

Ms. Pranali Pawar, Member

Dr. Hiren Dand , VC nominee

Professor (Dr.) Ajay S Patil, Subject Expert, North Maharashtra University

Mr. Milind Narayan Kolambe, Subject Expert, Pune University

Mr. Tejpal Khachane, Industry Expert

Mr. Abhishek Ghorpade, Postgraduate meritorious alumnus

Mrs. Pournima P Bhangale
Chairperson

Dr. Hiren Dand
VC Nominee