The Kelkar Education Trust's
V G Vaze College of Arts, Science and Commerce
(Autonomous)

Syllabus for F. Y. B.Sc. – I.T.

(June 2022 Onwards)

Programme: B.Sc.

Subject : Information Technology

Semester I & II

Semester I			
Course Code	**Course Type**	**Course Title**	**Credits**
SIT101	Core Subject	Imperative Programming	2
SIT102	Core Subject	Digital Electronics	2
SIT103	Core Subject	Microprocessor Architecture	2
SIT104	Core Subject	Discrete Mathematics	2
SIT105	Ability Enhancement Skill Course	Communication Skills	2
SITP101	Core Subject Practical	Imperative Programming Practical	2
SITP102	Core Subject Practical	Digital Electronics Practical	2
SITP103	Core Subject Practical	Microprocessor Architecture Practical	2
SITP104	Core Subject Practical	Discrete Mathematics Practical	2
SITP105	Ability Enhancement Skill Course Practical	Communication Skills Practical	2
		Total Credits	**20**

Semester II			
Course Code	**Course Type**	**Course Title**	**Credits**
SIT201	Core Subject	Object oriented Programming	2
SIT202	Core Subject	Database Management Systems	2
SIT203	Core Subject	Web Programming	2
SIT204	Core Subject	Numerical and Statistical Methods	2
SIT205	Ability Enhancement Skill Course	Green Computing	2
SITP201	Core Subject Practical	Object Oriented Programming Practical	2
SITP202	Core Subject Practical	Database Management Systems Practical	2
SITP203	Core Subject Practical	Web Programming Practical	2
SITP204	Core Subject Practical	Numerical and Statistical Methods Practical	2
SITP205	Ability Enhancement Skill Course Practical	Green Computing Practical	2
		Total Credits	**20**

SEMESTER I

B. Sc (Information Technology)		Semester – I	
Course Name: Imperative Programming		**Course Code: SIT101**	
Periods per week (1 Period is 50 minutes)		**5**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	**2**	**60**
	Internal	**--**	**40**

Course Objective
To make learner understand and use
1. Concept of algorithm, flowchart and pseudocode.
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 pre-processordirectives, structure and union in C.
5. Concept of pointers in C.

Unit	Details	Lectures
I	**Introduction:** Types of Programming languages, History, Features and application. Simple program logic, program development cycle, pseudocode statements and flowchart symbols, sentinel value to end a program, programming and user environments, evolution of programming models., desirable program characteristics. **Fundamentals:** Structure of a program. Compilation and Execution of a Program, Character Set, identifiers and keywords, datatypes, constants, variables and arrays, declarations, expressions, statements, Variable definition, symbolic constants. **Operators and Expressions:** Arithmetic operators, unary operators, relational and logical operators, assignment operators, assignment operators, the conditional operator, library Functions.	**12**
II	**Data Input and output:** Single character input and output, entering input data, scan function, print function, gets and puts Functions, interactive programming. **Conditional Statements and Loops:** Decision Making Within A Program, Conditions, Relational Operators, Logical Connectives, if Statement, if-else statement, loops: while loop, do while, for loop. nested loops, infinite loops, switch statement	**12**
III	**Arrays:** Definition, processing, multidimensional arrays, arrays and strings. **Functions:** Overview, defining function, accessing a function, passing arguments to a function, specifying argument data types, function prototypes, recursion, modular programming and functions, standard library of C functions, prototype of a function, return type, function call, block structure, passing arguments to a function: call by reference, call by value. passing arrays to functions	**12**

IV	**Program structure:** Storage classes, automatic variables, external variables, static variables, multi-file programs, more library functions **Preprocessor:** Features, #define and #include, Directives and Macros **Structures and Unions:** Structure Variables, Initialization, Structure Assignment, Nested Structure, Structures and Functions, Structures and Arrays: Arrays of Structures, Structures Containing Arrays, Unions**,** Structures and pointers.	**12**
V	**Pointers:** Fundamentals: declarations, Pointers Address Operators, Pointer Type Declaration, Pointer Assignment, Pointer Initialization, Pointer Arithmetic, Functions and Pointers, Arrays And Pointers, Pointer Arrays, passing Functions to other Functions.	**12**

Course Outcome	
Learners should be able to	
CO1	Draw flowchart for a given problem.
CO2	Write algorithm and pseudocode for a given problem.
CO3	Understand various concepts of C language.
CO4	Implement the C language Concepts.
CO5	Motivate them to develop projects / applications using C language.

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Programming with C	Byron Gottfried	Tata McGRAW-Hill	2nd	1996
2.	Programming Logic and Design	Joyce Farell	Cengage Learning	8th	2014
3.	"C" Programming"	Brian W. Kernighan and Denis M. Ritchie.	PHI	2nd	
4.	Let us C	Yashwant P. Kanetkar,	BPB publication		
5.	C for beginners	Madhusudan Mothe	X-Team Series	1st	2008
6.	21st Century C	Ben Klemens	OReilly	1st	2012
7.	Programming in ANSI	E. Balagurusamy	Tata McGRAW-Hill		

B. Sc (Information Technology)		Semester – I	
Course Name: Imperative Programming Practical		Course Code: SITP101	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2	50
	Internal	--	--

List of Practical: (Can be done in any imperative language)	
1.	**Basic Programs:**
a.	Write a program to display the message HELLO WORLD.
b.	Write a program to declare some variables of type int, loat and double. Assign some values to these variables and display these values.
c.	Write a program to find the addition, subtraction, multiplication and division o two numbers.
2.	**Programs on variables:**
a.	Write a program to swap two numbers without using third variable.
b.	Write a program to find the area of rectangle, square and circle.
c.	Write a program to find the volume of a cube, sphere, and cylinder.
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.
f.	Write a program to find the largest of three numbers.
4.	**Conditional statements and loops(advanced)**
a.	Write a program to find the sum of squares of digits of a number.
b.	Write a program to reverse the digits of an integer.
c.	Write a program to find the sum of numbers from 1 to 100.
d.	Write a programs to print the fibbonacci series.
e.	Write a program to find the reverse of a number.
f.	Write a program to find whether a given number is palindrome or not.
g.	Write a program that solve the quadratic equation $x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$
h.	Write a program to check whether the entered number is Armstrong or not.
i.	Write a program to count the digit in a number
5.	**Programs on patterns:**
a.	Programs on different patterns.

6.	**Functions:**
a.	Programs on Functions.
7.	**Recursive Functions**
a.	Write a program to find the factorial of a number using recursive function.
b.	Write a program to find the sum of natural number using recursive function.
8.	**Arrays**
a.	Write a program to find the largest value that is stored in the array.
b.	Write a program using pointers to compute the sum of all elements stored in an array.
c.	Write a program to arrange the 'n' numbers stored in the array in ascending and/or descending order.
d.	Write a program that performs addition and subtraction of matrices.
e.	Write a program that performs multiplication of matrices.
9.	**Pointers**
a.	Write a program to demonstrate the use of pointers.
b.	Write a program to perform addition and subtraction of two pointer variables.
10.	**Structures and Unions**
a.	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.
b.	Programs on unions.

B. Sc (Information Technology)		Semester – I	
Course Name: Digital Electronics		Course Code: SIT102	
Periods per week (1 Period is 50 minutes)		5	
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 number system, octal number system, hexadecimal number system, conversion from one number system to another, floating point numbers, weighted codes binary coded decimal, non-weighted codes Excess–3code, Gray code, Alphanumeric codes – ASCII Code, EBCDIC, ISCII Code, Hollerith Code, Morse Code, Teletypewriter (TTY), Error detection and correction, Universal Product Code, Code conversion. **Binary Arithmetic:** Binary addition, Binary subtraction, Negative number representation, Subtraction using 1's complement and 2's complement, Binary multiplication and division, Arithmetic in octal number system, Arithmetic in hexadecimal number system, BCD and Excess – 3 arithmetic.	12
II	**Boolean Algebra and Logic Gates:** Introduction, Logic (AND OR NOT), Boolean theorems, Boolean Laws, De Morgan's Theorem, Perfect Induction, Reduction of Logic expression using Boolean Algebra, Deriving Boolean expression from given circuit, exclusive OR and Exclusive NOR gates, Universal Logic gates, Implementation of other gates using universal gates, Input bubbled logic, Assertion level. **Minterm, Maxterm and Karnaugh Maps:** Introduction, minterms and sum of minterm form, maxterm and Product of maxterm form, Reduction technique using Karnaugh maps – 2/3/4/5/6 variable K-maps, Grouping of variables in K-maps, K-maps or product of sum form, minimize Boolean expression using K-map and obtain K-map from Boolean expression, Quine McCluskey Method.	12

III	**Combinational Logic Circuits:** Introduction, Multi-input, multi-output Combinational circuits, Code converters design and implementations **Arithmetic Circuits:** Introduction, Adder, BCD Adder, Excess – 3 Adder, Binary Subtractors, BCD Subtractor, Multiplier, Comparator.	**12**
IV	**Multiplexer, Demultiplexer, ALU, Encoder and Decoder:** Introduction, Multiplexer, Demultiplexer, Decoder, ALU, Encoders. **Sequential Circuits: flip-flop:** Introduction, Terminologies used, S-R flip-flop, D lip-op, JK flip-flop, Race-around condition, Master–slave JK flip-flop, T flip-flop, conversion from one type of flip-flop to another, Application of flip-flops.	**12**
V	**Counters:** Introduction, Asynchronous counter, Terms related to counters, IC 7493 (4-bit binary counter), Synchronous counter, Type T Design, Type JK Design, Synchronous counter ICs, Analysis of counter circuits. **Shift Register:** Introduction, parallel and shift registers, serial shifting, serial–in serial–out, serial–in parallel–out , parallel–in parallel–out, Ring counter, Johnson counter, Applications of shift registers, Pseudo-random binary sequence generator, IC7495, Seven Segment displays, analysis of shift counters.	**12**

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 – I	
Course Name: Digital Electronics Practical		Course Code: SITP102	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2	50
	Internal	--	--

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
c.	Implement AND, OR, NOT, XOR, XNOR using NAND gates.
d.	Implement AND, OR, NOT, XOR, XNOR using NOR gates.
2.	**Implement the given Boolean expressions using minimum number of gates.**
a.	Verifying De Morgan's laws.
b.	Implement other given expressions using minimum number of gates.
c.	Implement other given expressions using minimum number of ICs.
3.	**Implement combinational circuits.**
a.	Design and implement combinational circuit based on the problem given and minimizing using K-maps.
4.	**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
5.	**Implement Adder and Subtractor Arithmetic circuits.**
a.	Design and implement Hal adder and full adder.
b.	Design and implement BCD adder.
c.	Design and implement XS – 3 adder.
d.	Design and implement binary subtractor.
e.	Design and implement BCD subtractor.
.	Design and implement XS – 3 subtractor.
6.	**Implement Arithmetic circuits.**
a.	Design and implement a 2-bit by 2-bit multiplier.
b.	Design and implement a 2-bit comparator.
7.	**Implement Encode and Decoder and Multiplexer and Demultiplexers.**
a.	Design and implement 8:3 encoder.
b.	Design and implement 3:8 decoder.
c.	Design and implement 4:1 multiplexer. Study of IC 74153, 74157
d.	Design and implement 1:4 demultiplexer. Study of IC 74139
e.	Implement the given expression using IC 74151 8:1 multiplexer.
.	Implement the given expression using IC 74138 3:8 decoder.

8.	**Study of flip-flops and counters.**
a.	Study of IC 7473.
b.	Study of IC 7474.
c.	Study of IC 7476.
d.	Conversion of flip-flops.
e.	Design of 3-bit synchronous counter using 7473 and required gates.
.	Design of 3-bit ripple counter using IC 7473.
9.	**Study of counter ICs and designing Mod-N counters.**
a.	Study of IC 7490, 7492, 7493 and designing mod-n counters using these.
b.	Designing mod-n counters using IC 7473 and 7400 (NAND gates)
10.	**Design of shit registers and shit register counters.**
a.	Design serial – in serial – out, serial – in parallel – out, parallel – in serial – out, parallel – in parallel – out and bidirectional shit registers using IC 7474.
b.	Study of ID 7495.
c.	Implementation of digits using seven segment displays.

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 – II	
Course Name: Microprocessor Architecture		Course Code: SIT103	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2	60
	Internal	--	40

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

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 the 8155 Memory Segment, Illustrative Example: Designing Memory for the MCTS Project, Testing and Troubleshooting Memory Interfacing Circuit,8085-Based Single-Board microcomputer.	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:** 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	**12**
IV	**Stacks and Sub-Routines:** Stack, Subroutine, Restart, Conditional Call, Return Instructions, Advanced Subroutine concepts. **Code Conversion, BCD Arithmetic, and 16-Bit Data Operations:** BCD-to-Binary Conversion, Binary-to-BCD Conversion, BCD-to-Seven-Segment-LED Code Conversion, Binary-to-ASCII and ASCII-to-Binary Code Conversion, BCD Addition, BCD Subtraction, Introduction To Advanced Instructions and Applications, Multiplication, Subtraction With Carry.	**12**
V	**Software Development System and Assemblers:** Microprocessors-Based Software Development system, Operating System and Programming Tools, Assemblers and Cross-Assemblers, Writing Program Using Cross Assemblers. **Interrupts:** The 8085 Interrupt, 8085 Vectored Interrupts, Restart as S/W Instructions, Additional I/O Concepts and processes. **The Pentium and Pentium Pro microprocessors:** Introduction, Special Pentium registers, Memory management, Pentium instructions, Pentium Pro microprocessor, Special Pentium Profeatures.	**12**

Course Outcome	
Learners should be able to	
CO1	Acquire basic knowledge of microprocessor.
CO2	Understand the architecture of a microprocessor 8085 and other.
CO3	Carry out various code conversions.
CO4	Write 8085 based programs.
CO5	Motivated to pursue further study in the field of Embedded systems.

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 – II	
Course Name: Microprocessor Architecture Practical		**Course Code: SITP103**	
Periods per week (1 Period is 50 minutes)		**3**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Practical Examination**	**2**	**50**
	Internal	**--**	**--**

List of Practical	
1.	**Perform the following Operations related to memory locations.**
a.	Store the data byte 32H into memory location C200H.
b.	Exchange the contents of memory locations C200H and C201H.
2.	**Simple assembly language programs.**
a.	Subtract the contents of memory location C201H from the memory location C200H and place the result in memory location C002H.
b.	Subtract two 8-bit numbers.
c.	Addthe16-bit number in memory locations C200H and C201H to the16-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.
d.	Add the contents of memory locations C2001H and C201H and place the result in the memory locations C202Hand C203H.
e.	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.
f.	Find the l's complement of the number stored at memory location C200H and storethe complemented number at memory location C300H.
g.	Find the 2's complement of the number stored at memory location C200H and store the complemented number at memory location C300H.
3.	**Packing and unpacking operations.**
a.	Pack the two unpacked BCD numbers stored in memory locations C200H and C201H and store result in memory location 4300H. Assume the least significant digit is stored at C200H.
b.	Two digit BCD number is stored in memory location C200H. Unpack the BCD number and store the two digits in memory locations C300H and C301H such that memory location C300H 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 C200H and the series begins from memory location C201H. a. Consider the sum to be 8-bit number. So,ignore carries. Store the sum at memory location C300H. b. Consider the sum to be 16-bit number. Store the sum at memory locations C300H and C301H
b.	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.
c.	Divide 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.
d.	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
e.	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.
6.	**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 series of even 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 8 bit number so you can ignore carries and store the sum at memory location C250H.
c.	Calculate the sum of series of 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 16-bit. Store the sum at memory locations C300H and C301H.
d.	Find the square of the given numbers from memory location C100H and store the result from memory location C200H.
e.	Search the given byte in the list of 50 numbers stored in the consecutive memory locations and store the address of memory location in the memory locations C200H and C201H. Assume byte is in C register and starting address of the list is C300H. If byte is not found store 00 at C200H and C201H.
f.	Two decimal numbers six digits each, are stored in BCD package form. Each number occupies a sequence of byte in the memory. The starting address of first number is C300H. Write an assembly language program that adds the set of two numbers and stores the sum in the same format starting from memory location C200H.
g.	Add 2 arrays having ten 8-bit numbers each and generate a third array of result. It is necessary to add the first element of array 1 with the first element of array-2 and so on. The starting addresses of arrayl, array2 and array3 are C200H, C300H and C400H, respectively.

7.	**Assembly programs on memory locations.**
a.	Write an assembly language program to separate even numbers from the give n list of 50 numbers and store them in another list starting from C300H. Assume starting address of 50 number list is C200H.
b.	Write assembly language program with proper comments for the following: A block of data consisting of 406 bytes is stored in memory starting at C300H. This block is to be shifted (relocated) in memory from 3050H onwards. Do not shift the block or part of the block anywhere else in the memory.
c.	Add even parity to a string of 7-bit ASCII characters. The length of the string is in memory location C040H and the string itself begins in memory location C041H. Place even parity in the most significant bit of each character.
d.	A list of 50 numbers is stored in memory, starting at C200H. Find number of negative, zero and positive numbers from this list and store these results in memory locations C300H, C301H, and C302H respectively.
e.	Write an assembly language program to generate fibonacci number.
f.	Program to calculate the factorial of a number between 0 to 8.
8.	**String operations in assembly programs.**
a.	Write an 8085-assembly language program to insert a string of four characters from the tenth location in the given array of 50 characters.
b.	Write an 8085-assembly language program to delete a string of 4 characters from the tenth location in the given array of 50 characters.
c.	Multiplythe8-bitunsignednumberinmemorylocation C200Hbythe8-bitunsigned number in memory location C201H. Store the 8 least significant bits of the result in memory location C300H and the 8 most significant bits in memory location C301H.
d.	Divide the 16-bit unsigned number in memory locations C200H and C201H (most significant bits in C201H) by the B-bit unsigned number in memory location C300H store the quotient in memory location C400H and remainder in C401H.
e.	DAA instruction is not present. Write a sub routine which will perform the same task as DAA.
9.	**Calculations on memory locations.**
a.	To test RAM by writing '1' and reading it back and later writing '0' (zero) and reading it back. RAM addresses to be checked are C100H to C1FFH. In case of any error, it is indicated by writing 01H at port 10.
b.	Arrange an array of 8-bit unsigned no in descending order.
c.	Transfer ten bytes of data from one memory to another memory block. Source memory block starts from memory location C200H where as destination memory block starts from memory location C300H.
d.	Write a program to find the Square Root of an 8-bit binary number. The binary number is stored in memory location C200H and store the square root in C201H.
e.	Write a simple program to Split a HEX data into two nibbles and store it in memory.
10.	**Operations on BCD numbers.**
a.	Add two 4-digit BCD numbers in HL and DE register pairs and store result in memory locations, C300H and C301H. Ignore carry after 16 bits.
b.	Subtract the BCD number stored in E register from the number stored in the D Register.
c.	Write an assembly language program to multiply 2 BCD numbers.

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

The Kelkar Education Trust's

V G Vaze College of Arts, Science and Commerce

(Autonomous)

B. Sc. (Information Technology)		Semester – I	
Course Name: Discrete Mathematics		**Course Code: SIT104**	
Periods per week (1 Period is 50 minutes)		**5**	
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. Use division into cases in a proof.
3. Concept of Relations, graphs and trees.
4. Construct correct direct and indirect proofs.
5. Use of counting and probability concept.

Unit	Details	Lectures
I	**Introduction**: Variables, The Language of Sets, The Language of Relations and function **Set Theory**: Definitions and the Element Method of Proof, Properties of Sets, Disproofs, Algebraic Proofs, Boolean Algebras, Russell's Paradox and the Halting Problem. **The Logic of Compound Statements**: Logical form and Logical Equivalence, Conditional Statements, Valid and Invalid Arguments	**12**
II	**Quantified Statements**: Predicates and Quantified Statements, Statements with Multiple Quantifiers, Arguments with Quantified Statements **Elementary Number Theory and Methods of Proof**: Introduction to Direct Proofs, Rational Numbers, Divisibility, Division into Cases and the Quotient-Remainder Theorem, floor and Ceiling, Indirect Argument: Contradiction and Contraposition, Two Classical Theorems, Applications in algorithms.	**12**
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 homogenous recurrence relations with constant coefficients. general recursive definitions and structural induction. **Functions**: Functions Defined on General Sets, One-to-One and Onto, Inverse Functions, Composition of Functions, Cardinality with Applications to Computability	**12**
IV	**Relations**: Relations on Sets, Reflexivity, Symmetry, and Transitivity, Equivalence Relations, Partial Order Relations **Graphs and Trees**: Definitions and Basic Properties, Trails, Paths, and Circuits, Matrix Representations of Graphs, Isomorphism's of Graphs, Trees, Rooted Trees, Isomorphism's of Graphs, Spanning trees and shortest paths.	**12**

V	**Counting and Probability:** Introduction, Possibility Trees and the Multiplication Rule, Possibility Trees and the Multiplication Rule, Counting Elements of Disjoint Sets: The Addition Rule, The Pigeonhole Principle, Counting Subsets of a Set: Combinations, r-Combinations with Repetition Allowed, Probability Axioms and Expected Value, Conditional Probability, Bayes' formula, and Independent Events.	**12**

Course Outcome	
Learners should be able to	
CO1	Think analytically.
CO2	Have better reasoning abilities.
CO3	Prove mathematical properties using mathematical induction methods, study functions, spaces, and other mathematical structures using sequences and use of recursion.
CO4	Use relation, graphs and trees in various applications.
CO5	Use 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: Discrete Mathematics Practical		Course Code: SITP104	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2	50
	Internal	--	--

List of Practical: Write the programs or the following using SCILAB	
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.	**Counting**
a.	Sum rule principle
b.	Product rule principle
c.	Factorial
d.	Binomial coefficients
e.	Permutations
f.	Permutations with repetitions
g.	Combinations
h.	Combinations with repetitions
i.	Ordered partitions
j.	Unordered partitions
4.	**Probability Theory**
a.	Sample space and events
b.	Finite probability spaces
c.	Equiprobable spaces
d.	Addition Principle
e.	Conditional Probability
f.	Multiplication theorem or conditional probability
g.	Independent events
h.	Repeated trials with two outcomes
5.	**Graph Theory**
a.	Paths and connectivity
b.	Minimum spanning tree
c.	Isomorphism

6.	**Directed Graphs**
a.	Adjacency matrix
b.	Path matrix
7.	**Properties of integers**
a.	Division algorithm
b.	Primes
c.	Euclidean algorithm
d.	Fundamental theorem of arithmetic
e.	Congruence relation
f.	Linear congruence equation
8.	**Algebraic Systems**
a.	Properties of operations
b.	Roots of polynomials
9.	**Boolean Algebra**
a.	Basic definitions in Boolean Algebra
b.	Boolean algebra as lattices
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

B. Sc (Information Technology)		Semester – I	
Course Name: Communication Skills		Course Code: SIT105	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2	60
	Internal	--	40

Course Objective
To make learner understand / implement
1. Recognized seven C's of communication.
2. Importance of nonverbal communication.
3. Oral, written communication skills.
4. Appreciated role of body language and voice tone in effective communication.
5. Communicate the message in businesses in an effective and engaging way or the recipient.

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, Cross-cultural communication, Technology-enabled Business Communication	12
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	12
III	**Developing Oral Communication Skills or Business:** Meetings and Conferences, Group Discussions and Team Presentations, Team Briefing, **Understanding Specific Communication Needs:** Communication across functional Areas	12
IV	**Understanding Specific Communication Needs:** Corporate Communication, Persuasive Strategies in Business Communication, Ethics in Business Communication, Business Communication Aids	12
V	**Presentation Process:** Planning the presentations, executing the presentations, Impressing the audience by performing, Planning stage: Brainstorming, mind maps / concept maps, executing stage: chunking theory, creating outlines, Use of templates. Adding graphics to your presentation: Visual communication, Impressstage: use of font, colour, layout, Importance of practice and performance.	12

Course Outcome	
Learners should be able to	
CO1	Communicate effectively.
CO2	Will be able to improve understanding and build rapport with others.
CO3	Communicate in a group
CO4	Prepare and deliver presentation effectively.
CO5	Prepare Writing 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	Oxord University Press	Second	
2.	Proessional Communication	ArunaKoneru	Tata McGraw Hill		
3.	Strategies or improving your business communication	Pro. M. S. Rao	Shro publishers and distributors		2016
4.	Business Communication	Dr. Rishipal and Dr. Jyoti Sheoran	SPD		2014
5.	Graphics or Learning: Proven Guidelines or Planning, Designing, and Evaluating Visuals in Training Materials	Ruth C. Clark, Chopeta Lyons,	Peier, Wiley		2011
6.	Basic Business Communication: Skills or Empowering the Internet Generation	Lesikar Raymond V and Marie E. latley.	Tata McGraw-Hill	10th	2005
7.	Nonverbal Communication: Notes on the Visual Perception o Human Relations	Ruesh, Jurgen and Weldon Kees	University of Caliornia Press		1966
8.	Business Communication Today	Bovee,Courtl and L.; Thill, John V.	Pearson Education Ltd.		2015
9.	Communication Skills	Dr. NageshwarRaoDr. Rajendra P. Das	Himalaya Publishing House		

B. Sc (Information Technology)		Semester – I	
Course Name: Communication Skills Practical		Course Code: SITP105	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2	50
	Internal	--	--

List of Practical Questions:	
1.	Communication Origami, Guessing Game, Guessing the emotion
2.	Body Language, follow All Instructions, Effective feedback Skills
3.	The Name Game, Square Talk (Effective Communication), Room 101 (Influential and persuasive skills)
4.	Back to Back Communication, Paper Shapes (Importance of two-way communication), Memory Test(Presentation Skills)
5.	Exercises on Communication Principles
6.	Exercises on communication icebreakers
7.	Communication exercises For the following practical, Microsoft Office, Open Office, Libre Office or any other software suite can be used.
8.	Use of word processing tools or communication
9.	Use of spreadsheet tools or communication
10.	Use of presentation tools or communication

SEMESTER II

B. Sc. (Information Technology)		Semester – II	
Course Name: Object Oriented Programming		Course Code: SIT201	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2	60
	Internal	--	40

Course Objective
To make learner understand / implement
6. Basic Object Oriented Methodology and principles.
7. Concept of classes and objects, constructors and destructors.
8. Polymorphism and virtual functions.
9. Inheritance and exception handling.
10. Template, File handling and OOPs concepts using C++.

Unit	Details	Lectures
I	**Object Oriented Methodology**: Introduction, Advantages and Disadvantages of Procedure Oriented Languages, what is Object Oriented? What is Object Oriented Development? Object Oriented Themes, Benefits and Application of OOPS. **Principles of OOPS:** OOPS Paradigm, Basic Concepts of OOPS: Objects, Classes, Data Abstraction and Data Encapsulation, Inheritance, Polymorphism, Dynamic Binding, Message Passing	12
II	**Classes and Objects:** Simple classes (Class specification, class members accessing), Defining member functions, passing object as an argument, Returning object from functions, friend classes, Pointer to object, Array of pointer to object. **Constructors and Destructors:** Introduction, Default Constructor, Parameterized Constructor and examples, Destructors	12
III	**Polymorphism:** Concept of function overloading, overloaded operators, overloading unary and binary operators, overloading comparison operator, overloading arithmetic assignment operator, Data Conversion between objects and basic types, **Virtual Functions:** Introduction and need, Pure Virtual Functions, Static Functions, this Pointer, abstract classes, virtual destructors.	12
IV	**Program development using Inheritance:** Introduction, understanding inheritance, Advantages provided by inheritance, choosing the access specifier, Derived class declaration, derived class constructors, class hierarchies, multiple inheritance, multilevel inheritance, containership, hybrid inheritance. **Exception Handling:** Introduction, Exception Handling Mechanism, Concept of throw & catch with example	12

V	**Templates:** Introduction, Function Template and examples, Class Template and examples. **Working with Files:** Introduction, File Operations, Various File Modes, File Pointer and their Manipulation	**12**

Course Outcome	
Learners should be able touse C++ language to	
CO1	Implement the basic principles of OOPs
CO2	Implement Object Oriented Conceptslikeclasses , constructors, destructors etc.
CO3	Implement various types of inheritance, polymorphism and virtual function.
CO4	Implement exception and file handling, template.
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 Edition	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 Practical		Course Code: SITP201	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2	50
	Internal	--	--

List of Practical: To be implemented using object oriented language	
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 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 the given number is armStrong or not.WhereareadNo() will be private method.
d.	Write a program to demonstrate function definition outside class and accessing class members in function definition.
2.	**Using friend functions.**
a.	Write a friend function for adding the two complex numbers, using a single class
b.	Write a friend function for adding the two different distances and display its sum, using two classes.
c.	Write a friend function for adding the two matrix from two different classes and display its sum.
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 variable and static function.
4.	**Operator Overloading**
a.	Overload the operator unary(-) for demonstrating operator overloading.
b.	Overloadtheoperator+foraddingthetimingsoftwoclocks,Andalsopassobjects 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 the 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.	**String handling**
a.	String operations for string length, string concatenation
b.	String operations for string reverse, string comparison,
c.	Console formatting functions.
8.	**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.
9.	**File handling**
a.	Design a class FileDemo open a file in read mode and display the total number of words and lines in the file.
b.	Design a class to handle multiple files and file operations
c.	Design a editor for appending and editing the files
10.	**Templates**
a.	Show the implementation for the following
b.	Show the implementation of template class library for swap function.
c.	Design the template class library for sorting ascending to descending and vice-Versa

The Kelkar Education Trust's

V G Vaze College of Arts, Science and Commerce

(Autonomous)

B. Sc. (Information Technology)		Semester – III	
Course Name: Database Management Systems		Course Code: SIT202	
Periods per week (1 Period is 50 minutes)		5	
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. Will study relational database, relational algebra and calculus data models and design the database schema with the use of appropriate data types for storage of data in database.
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 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, ER Design 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 andHierarchical Retrieval, Triggers.	**12**

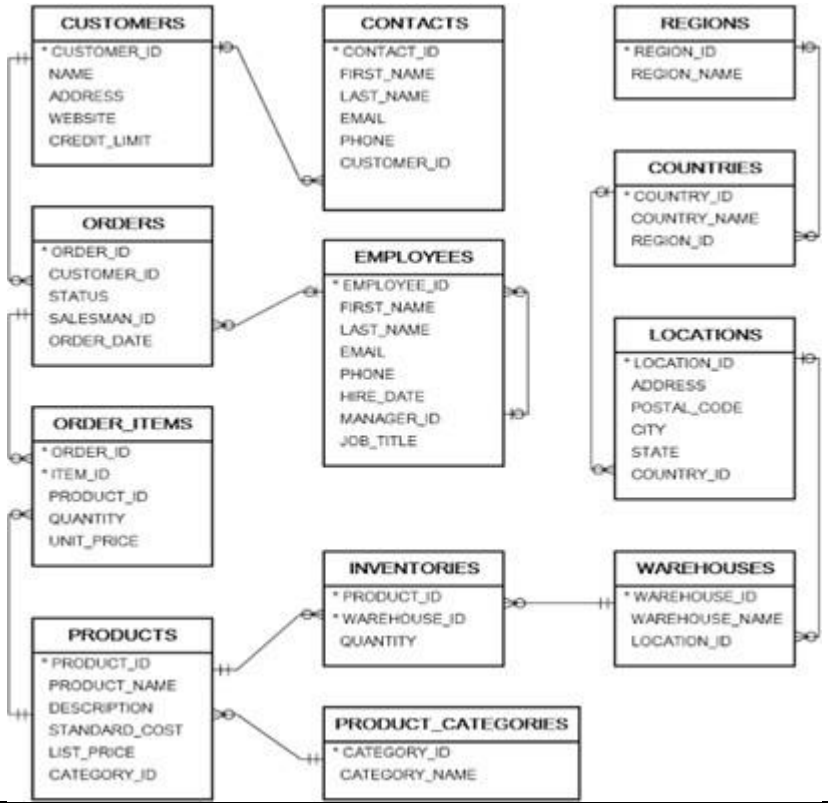
Course Outcome	
Learnerwill be able to	
CO1	Understand the importance of database management system.
CO2	Create database schema using ER model and Relational Model.
CO3	Write various SQL queries, apply various constraints, creating database objects.
CO4	Apply PL SQL concepts.
CO5	Motivate the student 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 Sudarshan	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 – III	
Course Name: Database Management System		Course Code: SITP202	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2	50
	Internal	--	--

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 ststement 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
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

The Kelkar Education Trust's

V G Vaze College of Arts, Science and Commerce (Autonomous)

B. Sc (Information Technology)		Semester – II	
Course Name: Web Programming		Course Code: SIT203	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2	60
	Internal	--	40

Course Objective
To make learner understand / develop
1. Basics of Internet and HTML5.
2. HTML5 elements.
3. Concept of JavaScript.
4. Concept of PHP and MYSQL.
5. Web pages using various concepts of HTML5, JavaScript, PHP and MYSQL.

Unit	Details	Lectures
I	**Internet and the World Wide Web:** What is Internet? Introduction to internet and its applications, E-mail, telnet, FTP, e-commerce, video conferencing, e-business. Internet service providers, domain name server, internet address, World Wide Web (WWW): World Wide Web and its evolution, uniform resource locator(URL), browsers–internet explorer, Netscape navigator, opera, Firefox, chrome, Mozilla. Search engine, websever apache, IIS, proxy server, HTTP protocol **HTML5:** Introduction, Why HTML5? Formatting text by using tags, using lists and backgrounds, Creating hyperlinks and anchors. Style sheets, CSS formatting text using style sheets, formatting paragraphs using style sheets.	12
II	**HTML5 Page layout and navigation:** Creating navigational aids: planning site organization, creating text based navigation bar, creating graphics based navigation bar, creating graphical navigation bar, creating image map, redirecting to another URL, creating division based layouts: HTML5 semantic tags, creating divisions, creating HTML5 semantic layout, positioning and formatting divisions. **HTML5 Tables, Forms and Media:** Creating tables: creating simple table, specifying the size of the table, specifying the width of the column, merging table cells, using tables for page layout, formatting tables: applying table borders, applying background and foreground fills, changing cell padding, spacing and alignment, creating user forms: creating basic form, using check boxes and option buttons, creating lists, additional input types in HTML5, Incorporating sound and video: audio and video in HTML5, HTML multimedia basics, embedding video clips, incorporating audio onwebpage.	12

III	**Java Script:** Introduction, Client-Side JavaScript, Server-Side JavaScript, JavaScript Objects, JavaScript Security, **Operators**: Assignment Operators, Comparison Operators, Arithmetic Operators, % (Modulus), ++(Increment), --(Decrement), -(Unary Negation), Logical Operators, Short-Circuit Evaluation, String Operators, Special Operators, ?: (Conditional operator), , (Comma operator), delete, new, this, void **Statements:** Break, comment, continue, delete, do...while, export, for, for...in, function, if...else, import, labelled, return, switch, var, while, with, **Core JavaScript (Properties and Methods of Each) :** Array, Boolean, Date, Function, Math, Number, Object, String, regExp **Document and its associated objects:** document, Link, Area, Anchor, Image, Applet, Layer **Events and Event Handlers :** General Information about Events, Defining Event Handlers, event, onAbort, onBlur, onChange, onClick, onDblClick, onDragDrop, onError, onFocus, onKeyDown, onKeyPress, onKeyUp, onLoad, onMouseDown, onMouseMove, onMouseOut, onMouseOver, onMouseUp, onMove, onReset, onResize, onSelect, onSubmit, onUnload	**12**
IV	**PHP:** Why PHP and MySQL? Server-side scripting, PHP syntax and variables, comments, types, control structures, branching, looping, termination, functions, passing information with PHP, GET, POST, formatting form variables, super global arrays, strings and string functions, regular expressions, arrays, number handling, basic PHP errors/problems	**12**
V	**Advanced PHP and MySQL:** PHP/MySQL Functions, Integrating web forms and databases, Displaying queries in tables, Building Forms from queries, String and Regular Expressions, Sessions, Cookies and HTTP, E-Mail	**12**

Course Outcome	
Learners should be able to	
CO1	Design web pages using HTML 5 elements.
CO2	Develop dynamic web pages using JavaScript.
CO3	Develop web applications using PHP and MySQL.
CO4	Develop static / dynamic web applications using HTML5, PHP and MYSQL.
CO5	Feel interested and motivated to pursue further study in the field of web development.

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Web Design The Complete Reference	Thomas Powell	Tata McGraw Hill		-
2.	HTML5 Step by Step	FaitheWempen	Microsoft Press		2011
3.	PHP 5.1 for Beginners	Ivan Bayross Sharanam Shah,	SPD		2013
4.	PHP Project for Beginners	SharanamShah, Vaishali Shah	SPD		2015
5.					
6.	PHP 6 and MySQL Bible	Steve Suehring, Tim Converse, Joyce Park	Wiley		2009
7.	Head First HTML 5 programming	Eric Freeman	O'Reilly		2013
8.	JavaScript 2.0: The Complete Reference	Thomas Powell and Fritz Schneider	Tata McGraw Hill	2nd	

B. Sc. (Information Technology)		Semester – II	
Course Name: Web Programming Practical		Course Code: SITP203	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2	50
	Internal	--	--

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 demonstrating all Style sheet types
2.	**Image maps, Tables, Forms and Media**
a.	Design a web page with Imagemaps.
b.	Design a web page demonstrating different semantics
c.	Design a web page with different tables. Design a webpages using table so that the content appears well placed.
d.	Design a web page with a form that uses all types of controls.
e.	Design a web page embedding with multimedia features.
3.	**Java Script**
a.	Using JavaScript design, a web page that prints factorial/Fibonacci series/any given series.
b.	Design a form and validate all the controls placed on the form using Java Script.
c.	Write a JavaScript program to display all the prime numbers between 1 and 100.
a.	Write a JavaScript program to accept a number from the user and display the sum of its digits.
d.	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).
e.	Write a java script program to design simple calculator.
4.	**Control and looping statements and Java Script references**
a.	Design a web page demonstrating different conditional statements.
b.	Design a web page demonstrating different looping statements.
c.	Design a web page demonstrating different Core JavaScript references (Array, Boolean, Date, Function, Math, Number, Object, String, regExp).
5.	**Basic PHP I**
a.	Write a PHP Program to accept a number from the user and print it factorial.
b.	WriteaPHPprogramtoacceptanumberfromtheuserandprintwhetheritisprime or not.
6.	**Basic PHP II**
a.	Write a PHP code to find the greater of 2 numbers. Accept the no. from the user.

b.	Write a PHP program to display the following Binary Pyramid: 1 0 1 1 0 1 0 1 0 1
7.	**String Functions and arrays**
a.	Write a PHP program to demonstrate different string functions.
b.	Write a PHP program to create one dimensional array.
8.	**PHP and Database**
a.	Write a PHP code to create: • Create a database College. • Create a table Department (Dname, Dno,Number_Of_faculty)
b.	Write a PHP program to create a database named "College". Create a tablenamed "Learner" with following fields (sno, sname, percentage). Insert 3 records of your choice. Display the names of the Learners whose percentage is between 35 to60 in a tabular format.
c.	Design a PHP page for authenticating a user.
9.	**Email**
a.	Write a program to send email with attachment.
10.	**Sessions and Cookies**
a.	Write a program to demonstrate use of sessions and cookies.

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	HTML5 Step by Step	FaitheWempen	Microsoft Press		2011
2.	JavaScript 2.0: The Complete Reference	Thomas Powell and Fritz Schneider	Tata McGraw Hill	2nd	
3.	PHP 6 and MySQL Bible	Steve Suehring, Tim Converse, Joyce Park	Wiley		2009
4.	PHP 5.1 for Beginners	Ivan Bayross Sharanam Shah,	SPD		2013
5.	PHP Project for Beginners	SharanamShah, Vaishali Shah	SPD		2015
6.	Murach's PHP and MySQL	Joel Murach Ray Harris	SPD		2011

B. Sc. (Information Technology)		Semester – II	
Course Name: Numerical and Statistical Methods		Course Code: SIT204	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2	60
	Internal	--	40

Course Objective

To make learner understand / develop

1. A concrete idea of what numerical methods are and how they relate to engineering and scientific problem solving as well as handling errors.
2. Different methods to solve algebraic and Transcendental equations as well as interpolation concept.
3. The concept of Numerical differentiation and Integration as well as Numerical solution of 1st and 2nd order differential equations
4. The least-square regression methods and linear programming.
5. The concept of random variables and distributions.

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. **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.	12
IV	**Least-Squares Regression:** Linear Regression, Polynomial Regression, Multiple Linear Regression, General Linear Least Squares, Nonlinear Regression **Linear Programming:** Linear optimization problem, Formulation and Graphical solution, Basic solution and Feasible solution.	12

V	**Random variables:** Discrete and Continuous random variables, Probability density function, Probability distribution of random variables, Expected value, Variance. **Distributions:** Discrete distributions: Uniform, Binomial, Poisson, Bernoulli, Continuous distributions: uniform distributions, exponential, (derivation of mean and variance only and state other properties and discuss their applications) Normal distribution state all the properties and its applications.	**12**

Course Outcome	
Learners should be able to	
CO1	Describe and discuss the key terminology, concepts tools and techniques used in business statistical analysis.
CO2	Critically evaluate the underlying assumptions of analysis tools.
CO3	Understand and critically discuss the issues surrounding sampling and significance.
CO4	To understanding of Statistical applications in Economics and Management and to deal with numerical and quantitative issues in business
CO5	Implement various techniques using Scilab.

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Introductory Methods of Numerical Methods	S. S. Shastri	PHI	Vol – 2	
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.	Fundamentals of Mathematical Statistics	S. C. Gupta, V. K. Kapoor			
5.	Elements of Applied Mathematics	P.N.Wartikar and J.N.Wartikar	A. V. Griha, Pune	Volume 1 and 2	

B. Sc. (Information Technology)		Semester – II	
Course Name: Numerical and Statistical Methods Practical		Course Code: SITP204	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2	50
	Internal	--	--

List of Practical	
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 usingSimpson'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**
a.	Program for Linear regression.
b.	Program for Polynomial Regression.

c.	Program for multiple linear regression.
d.	Program for non-linear regression.
9.	**Random variables and distributions**
a.	Program to generate random variables.
b.	Program to fit binomial distribution.
c.	Program to fit Poisson distribution.
10.	**Distributions**
a.	Program for Uniform distribution.
b.	Program for Bernoulli distribution
c.	Program for Negative binomial distribution.

B. Sc. (Information Technology)		Semester – II	
Course Name: Green Computing		Course Code: SIT205	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2	60
	Internal	--	40

Course Objective
The learner should understand
1. Meaning of Green Computing as well as different standards.
2. The modern approaches in Green Computing
3. General tactics to make computing usage greener.
4. How to prepare action plan for Green Computing.
5. The use of methods and tools to measure 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.	12
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. **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, Put Everything Together.	12

III	**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:** Problems, China, Africa, Materials, 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, In Practice	**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**

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	Discuss how the choice of hardware and software can facilitate a more sustainable operation.
CO5	Implement Green IT in Real Life.

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Green IT	Toby Velte, Anthony Velte, Robert Elsenpeter	McGraw Hill		2008
2.	Green Data Center: Steps for the Journey	Alvin Galea, Michael Schaefer, Mike Ebbers	Shroff Publishers and Distributers		2011
3.	Green Computing and Green IT Best Practice	Jason Harris	Emereo		
4.	Green Computing Tools and Techniques for Saving Energy, Money and Resources	Bud E. Smith	CRC Press		2014

B. Sc. (Information Technology)		Semester – II	
Course Name: Green Computing Practical		Course Code: SITP205	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2	50
	Internal	--	--

Project and Viva Voce	
1.	AprojectshouldbedonebasedontheobjectivesofGreenComputing.Areport of minimum 50 pages should be prepared. The report should have a font size of12,Timesnewromanand1.5linespacing.Theheadingsshouldhavefont size 14. The report should be hard bound.
2.	The project can be done individually or a group of two Learners.
3.	The Learners will have to present the project during the examination.
4.	A certified copy of the project report is essential to appear for the examination.

Evaluation Schem

1. **Internal Evaluation (40Marks).**
   i. **Test: 1 Class test of 15 marks. (Can be takenonline)**

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

   ii. **15 marks project / presentation**
   iii. **10 marks: Active participation in the class, overall conduct,attendance.**

2. **External Examination: (60marks)**

	All questions are compulsory	
Q1	(Based on whole syllabus) Attempt ***any two*** *of* the following:	10
a.		
b.		
c.		
d.		
Q2	(Based on Unit 1) Attempt ***any two*** *of* the following:	10
Q3	(Based on Unit 2) Attempt ***any two*** *of* the following:	10
Q4	(Based on Unit 3) Attempt ***any two*** *of* the following:	10
Q5	(Based on Unit 4) Attempt ***any two*** *of* the following:	10
Q6	(Based on Unit 5) Attempt ***any two*** *of* the following:	10

3. **Practical Exam: 50marks**

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

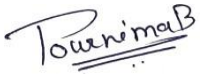
1.	Practical Question 1	20
2.	Practical Question 2	20
3.	Journal	5
4.	Viva Voce	5

OR

1.	Practical Question	40
2.	Journal	5
3.	Viva Voce	5

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
Professor (Dr.) Ajay S Patil,Subject Expert, North Maharashtra University
Dr. Rajendra B Patil, Subject Expert, VC Nominee
Mr. Yogesh Tambe, Industry Representative
Mr. Gandhar Patwardhan, Alumni Representative



Mrs. Pournima P Bhangale
Chairperson

Dr. Rajendra B Patil,
Subject Expert, VC Nominee