Vivekanand Education Society's

College of Arts, Science and Commerce

(Autonomous)

Sindhi Society, Chembur, Mumbai, Maharashtra – 400 071.

Accredited by NAAC "A Grade" in 3rd Cycle - 2017
Best College Award – Urban Area, University of Mumbai (2012-13)
Recipient of FIST Grant (DST) and STAR College Grant (DBT)

Affiliated to the

University of Mumbai

Syllabus for

Program: B.Sc. (Information Technology)

(Program code: VESUSIT)

As per Choice Based Semester and Grading System (CBSGS)
with effect from Academic Year 2022 - 2023

Program Outcomes (PO):

A leaner completing B.Sc.(Information Technology) will be able to:

PO1 To think analytically, creatively and critically in developing robust, extensible and highly maintainable technological solutions to simple and complex problems

PO2 To apply their knowledge and skills to be employed and excel in IT professional careers and/or to continue their education in IT and/or related postgraduate programmes.

PO3 To work effectively as a part of a team to achieve a common stated goal.

PO4 To communicate effectively in English

PO5 Be Environment Sensitive

Program Specific Outcomes (PSO's)

On completion of B.Sc.(Information Technology)program, learners will be enriched with knowledge and be able to

PSO1 Develop Desktop and Web applications using high level languages and scripting languages such as C, C++, JAVA, PHP,ASP.Net

PSO2 Know the functions of various modules of different types of operating systems.

PSO3 Design, code test and Document a software

PSO4 Have knowledge about different networking and security concepts

POS5 Understanding of latest technologies like Internet of things ,Artificial Intelligence and Geographic Information Systems and making applications using them

F.Y.B.Sc. Information Technology

(SEMESTER I)

Course Code	Title	Credits & Lectures per Semester	Lectures per Week
VESUSIT101	**Imperative Programming**	**02**	**05**
	Unit I :	**12 Lectures**	
	Unit II:	**12 Lectures**	
	Unit III :	**12 Lectures**	
	Unit IV:	**12 Lectures**	
	Unit V :	**12 Lectures**	
VESUSIT102	**Digital Electronics**	**02**	**05**
	Unit I :	**12 Lectures**	
	Unit II:	**12 Lectures**	
	Unit III :	**12 Lectures**	
	Unit IV:	**12 Lectures**	
	Unit V :	**12 Lectures**	
VESUSIT103	**Operating Systems**	**02**	**05**
	Unit I :	**12 Lectures**	
	Unit II:	**12 Lectures**	
	Unit III :	**12 Lectures**	
	Unit IV:	**12 Lectures**	
	Unit V :	**12 Lectures**	
VESUSIT104	**Discrete Mathematics**	**02**	**05**
	Unit I :	**12 Lectures**	
	Unit II:	**12 Lectures**	
	Unit III :	**12 Lectures**	
	Unit IV:	**12 Lectures**	

	Unit V :	**12 Lectures**	
VESUSIT105	**Communication Skills**	**02**	**05**
	Unit I :	**12 Lectures**	
	Unit II:	**12 Lectures**	
	Unit III :	**12 Lectures**	
	Unit IV:	**12 Lectures**	
	Unit V :	**12 Lectures**	
VESUSITP101	Imperative Programming Practical	**02**	**03**
VESUSITP102	Digital Electronics Practical	**02**	**03**
VESUSITP103	Operating Systems Practical	**02**	**03**
VESUSITP104	Discrete Mathematics Practical	**02**	**03**
VESUSITP105	Communication Skills Practical	**02**	**03**

F.Y.B.Sc. Information Technology

(SEMESTER II)

Course Code	Title	Credits & Lectures per Semester	Lectures per Week
VESUSIT201	Object oriented Programming	02	05
	Unit I :	12 Lectures	
	Unit II:	12 Lectures	
	Unit III :	12 Lectures	
	Unit IV:	12 Lectures	
	Unit V :	12 Lectures	
VESUSIT202	Microprocessor Architecture	02	05
	Unit I :	12 Lectures	
	Unit II:	12 Lectures	
	Unit III :	12 Lectures	
	Unit IV:	12 Lectures	
	Unit V :	12 Lectures	
VESUSIT203	Web Programming	02	05
	Unit I :	12 Lectures	
	Unit II:	12 Lectures	
	Unit III :	12 Lectures	
	Unit IV:	12 Lectures	
	Unit V :	12 Lectures	
VESUSIT204	Numerical and Statistical Methods	02	05
	Unit I :	12 Lectures	
	Unit II:	12 Lectures	
	Unit III :	12 Lectures	

	Unit IV:	**12 Lectures**	
	Unit V :	**12 Lectures**	
VESUSIT205	**Green Computing Methods**	**02**	**05**
	Unit I :	**12 Lectures**	
	Unit II:	**12 Lectures**	
	Unit III :	**12 Lectures**	
	Unit IV:	**12 Lectures**	
	Unit V :	**12 Lectures**	
VESUSITP201	**Object Oriented Programming Practical**	**02**	**03**
VESUSITP202	**Microprocessor Architecture Practical**	**02**	**03**
VESUSITP203	**Web Programming Practical**	**02**	**03**
VESUSIT204	**Numerical and Statistical Methods Practical**	**02**	**03**
VESUSITP205	**Green Computing Practical**	**02**	**03**

Detailed Syllabus: Unit wise / Module wise with number of lectures

Course title: Imperative Programming
Course code: VESUSIT101

Objective: To understand and develop programs using C Language

Learning Outcomes (LO):

On successful completion of this course students will be able to:

LO1	Develops the ability to study a problem, write an algorithm to solve it.
LO2	Develop basics of programming skills using C Language.
LO3	Develop programs using the control statements, Arrays and Strings
LO4	Can make effective usage of arrays, structures, functions and pointers.

Unit no.	Details of topics	No of lectures
1	**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. BG:1.4,1.7 **Fundamentals:** Structure of a program. Compilation and Execution of a Program, Character Set, identifiers and keywords, data types, constants, variables and arrays, declarations, expressions, statements, Variable definition,symbolic constants. BG:1.5,2.1,2.2,2.3,2.4,2.5,2.6,2.7,2.8,2.9	12 Lectures
2	**Operators and Expressions:** Arithmetic operators, unary operators, relational and logical operators,assignment operators, assignment operators, the conditional operator,library functions. BG:3.1,3.2,3.3,3.4,3.5,3.6 **Data Input and output:** Single character input and output, entering input data, scanf function,printf function, gets and puts functions, interactive programming. BG:4.2,4.3,4.4,4.6,4.8,4.9	12 Lectures
3	**Conditional Statements and Loops:**	15

	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 BG:3.3.6.2,6.3,6.4,6.5,6.66.7,6.8,6.9 Functions: Overview, defining a 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: foo1lal parameter list, return type, function call, block structure, passing arguments to a function: call by reference, call by value. BG:7.2,7.3,7.4,7.5,7.6,	Lectures
4	**Program structure:** Storage classes, automatic variables, external variables, static variables,multi file programs, more library functions, BG:8.1,8.2,8.3,8.4,8.5,8.6 Preprocessor: Features, #define and #include, Directives and Macros BG:14.5 **Arrays:** Definition, processing, passing arrays to functions, multidimensional arrays, arrays and strings. BG:9.1,9.2,9.3,9.4,9.5	
5	**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 BG:10.1,10.2,10.3,10.4,10.5,10.6,10.7,10.8,10.9,10.10 **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. BG:11.1,11.2,11.3,11.4,11.5,11.6.11.7	

References:

BG: Programming with C By Byron Gottfried Tata McGRAW-Hill 2nd Ed 1996

Additional references:

Programming Logic and Design By Joyce Farell Cengage Learning 8th Ed 2014
“C” Programming” Brian W.Kernighan and Denis M. Ritchie.PHI 2nd Ed
Let us C Yashwant P.Kanetkar,BPB Publication
C for beginners Madhusudan Mothe X-Team Series 1St Ed 2008
21 st Century C Ben Klemens OReilly 1st Ed 2012

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, float and double. Assign some values to these variables and display these values.
c.	Write a program to find the addition, subtraction, multiplication and division of 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 Fibonacci 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 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.	Programs on structures.
b.	Programs on unions.

Course title: Digital Electronics

Course code: VESUSIT102

Objective: To understand basics about Digital Electronics knowledge and develop analysis and design of digital electronic circuits

Learning Outcomes (LO):

On successful completion of this course students will be able to:

LO1 The course helps in design and analysis of the digital circuit and system.
LO2: Apply and solve the concept of sequential circuits for various applications like counters,shift registers etc.
LO3: Digital circuit design by Boolean expression using Boolean algebra and using logic gates
LO4: Analyze and design combinational circuits like code converter, multiplexers etc.

Unit	Details of topics	No.of 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 – 3 code, Gray code, Alphanumeric codes – ASCII Code, EBCDIC, ISCII Code, ALU Ic 74181,Universal Product Code, Code conversion. RJ 2.1-2.4.ML 2.1-2.4.2.8-2.10,3.7-3.8 **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. RJ 2.4-2.10 .ML 2.2,2.5	**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. RJ 1.1-1.6 . ML4.1-4.10 **Minterm, Maxterm and Karnaugh Maps:** Introduction, minterms and sum of minterm form, maxterm and Product of maxterm form, Reduction technique using Karnaugh maps – 2/3/4/5/6 variable K-maps, Grouping of variables in K-maps, K-maps for product of sum form, minimize Boolean expression using K-map and obtain K-map from Boolean expression, Applications of K map technique RJ 5.1-5.10 ML 6.1-6.11	**12**
III	**Combinational Logic Circuits:** Introduction, Multi-input, multi-output Combinational circuits, Code converters design and implementations RJ 6.1,6.9 **Arithmetic Circuits:** Introduction, Adder, BCD Adder, Excess – 3 Adder, Binary Subtractors, BCD Subtractor, Multiplier, Comparator. RJ 6.4-6.5 ML 5.1-5.6	**12**
IV	**Multiplexer, Demultiplexer, ALU, Encoder and Decoder:** Introduction, Multiplexer, Demultiplexer, Decoder, ALU, Encoders. **RJ 6**.2-6.3.6.6 ML 10.6-10.8 **Sequential Circuits: Flip-Flop:** Introduction, Terminologies used, S-R flip-flop, D flip-fop, 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. RJ 7.1-7.10 ML 7.1-7.5	**12**

V	**Counters:** Introduction, Asynchronous counter, Terms related to counters, IC 7493 (4-bit binary counter), Synchronous counter, Bushing, Type T Design, Type JK Design, Presettable counter, IC 7490, IC 7492, Synchronous counter ICs, Analysis of counter circuits. RJ 8.4-8.5 ML 8.1-8.6 **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,.TTL and CMOS logic families and characteristics,IC fabrication technology. RJ 8.1-8.2 .4.8-4.14 ML 9.1,9.6 ,1.6-1.7	**12**

References:

RJ:Modern Digital Electronics.Third Edition by R.P.Jain.Tata McGraw Hill publishing company Limited
ML:Digital principles and applications Third Edition by AlbertPaul Malvino and Donald p Leach Tata McGraw Hill publishing company Limited

Additional References:

Digital Electronics and Logic Design by N. G. Palan. Technova publisher
Digital Electronics: Principles, Devices and Applications by Anil K. Maini Wiley publisher

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 Half adder and Full adder.
b.	Design and implement BCD adder.
c.	Design and implement XS – 3 adder.
d.	Design and implement binary subtractor.
e.	Design and implement BCD subtractor.
f.	Design and implement XS – 3 subtractor.
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.
f.	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.
f.	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 shift registers and shift register counters.**
a.	Design serial – in serial – out, serial – in parallel – out, parallel – in serial – out, parallel – in parallel – out and bidirectional shift registers using IC 7474.
b.	Study of ID 7495.
c.	Implementation of digits using seven segment displays.

Course Title : Operating System
Course Code : VESUSIT103

Objective:

To analyze and comprehend the core of the operating system including the various management and processes involved in it.

Learning Outcomes (LO):

On successful completion of this course students will be able to:

LO 1 Comprehend and evaluate the role of operating systems in their management policies and algorithms.

LO 2 To learn different process scheduling algorithms and synchronization techniques to achieve better performance of a computer system.

LO 3 Analyze essential properties of different types of operating systems and their services.

LO 4 Awareness of Android and Linux Operating System

Unit No	Details of topics	No of Lectures
I	**Introduction:** What is an operating system? History of operating system, computer hardware, different operating systems, operating system concepts, system calls, operating system structure. AT: 1.1 -1.7 **Processes and Threads:** Processes, threads, interprocess communication, Methods of IPC, Communication in process using signals, scheduling, IPC problems. AT: 2.1 - 2.2	**12**
II	**Memory Management:** No memory abstraction, memory abstraction: address spaces, virtual memory, page replacement algorithms, design issues for paging systems, implementation issues, segmentation. AT: 3.1 - 3.7	**12**

III	**File Systems:** Files, directories, file system implementation, file-system management, and optimization, MS-DOS file system, UNIX V7 file system, CD ROM file system. AT: 4.1 - 4.5 **Input-Output:** Principles of I/O hardware, Principles of I/O software, I/O software layers, disks, clocks, user interfaces: keyboard, mouse, monitor, thin clients, power management. AT: 5.1 - 4.8	**12**
IV	**Virtualization and Cloud:** History, requirements for virtualization, type 1 and 2 hypervisors, techniques for efficient virtualization, hypervisor microkernels, memory virtualization, I/O virtualization, Virtual appliances, virtual machines on multicore CPUs, Clouds. AT: 7.1 - 7.11 **Multiple Processor Systems** Multiprocessors, microcomputers, distributed systems. AT: 8.1 - 8.3	**12**
V	**Deadlocks:** Resources, introduction to deadlocks, the ostrich algorithm, deadlock detection and recovery, deadlock avoidance, deadlock prevention, Issues. AT: 6.1 - 6.7 **Case Study on LINUX:** History of Unix and Linux, Linux Overview, Processes in Linux, Memory management in Linux, I/O in Linux, Linux file system, security in Linux. AT: 10.1 - 10.8 **Case Study on Windows:** History of windows through Windows 10, programming windows, system structure, processes and threads in windows, memory management, caching in windows, I/O in windows, Windows NT file system, Windows power management, Security in windows. AT: 11.1 - 11.10	**12**

References :

AT : Modern Operating Systems by Andrew S.Tanenbaum, Herbert Bos, Pearson, 4th Edition

Additional References:

Operating Systems – Internals and Design Principles by Willaim Stallings, Pearson, 8th Edition,
Operating System Concepts by Abraham Silberschatz, Peter B.Galvineg Gagne, By Wiley 8th Edition.
Operating Systems by Godbole and Kahate, McGraw Hill, 3rd Edition

List of Practical	
1.	Installation of virtual machine software.
2.	Installation of Linux operating system (RedHat / Ubuntu) on virtual machine.
3.	Installation of Windows operating system on virtial machine.
4.	**Linux commands: Working with Directories:**
a.	pwd, cd, absolute and relative paths, ls, mkdir, rmdir,
b.	file, touch, rm, cp. mv, rename, head, tail, cat, tac, more, less, strings, chmod
5.	**Linux commands: Working with files:**
a.	ps, top, kill, pkill, bg, fg,
b.	grep, locate, find, locate.
c.	date, cal, uptime, w, whoami, finger, uname, man, df, du, free, whereis, which.
d.	Compression: tar, gzip.
6.	**Windows (DOS) Commands – 1**
a.	Date, time, prompt, md, cd, rd, path.
b.	Chkdsk, copy, xcopy, format, fidsk, cls, defrag, del, move.
7.	**Windows (DOS) Commands – 2**
a.	Diskcomp, diskcopy, diskpart, doskey, echo
b.	Edit, fc, find, rename, set, type, ver
8.	**Working with Windows Desktop and utilities**
a.	Notepad
b.	Wordpad
c.	Paint
d.	Taskbar
e.	Adjusting display resolution

f.	Using the browsers
g.	Configuring simple networking
h.	Creating users and shares
9.	**Working with Linux Desktop and utilities**
a.	The vi editor.
b.	Graphics
c.	Terminal
d.	Adjusting display resolution
e.	Using the browsers
f.	Configuring simple networking
g.	Creating users and shares
10.	**Installing utility software on Linux and Windows**

Course Title : Discrete Mathematics
Course Code : VESUSIT104

Objective:
To develop the logical and analytical thinking of the student which will familiarize them with the concepts required for the competitive exams

Learning Outcomes (LO):

On successful completion of this course students will be able to:

LO1: Use concepts of Mathematical Logic , Conditional statements , and identify valid and invalid arguments
LO2: Explain the importance of quantified statements and describe sequences, mathematical induction and recursion.
LO3: Classify relations, graphs and trees ,implement functions on general sets and solve problems related to counting and probability.

Unit	Details	Lectures
I	**Introduction**: Variables, The Language of Sets, The Language of Relations and Function **DMA 1.1-1.3** **Set Theory**: Definitions and the Element Method of Proof, Properties of Sets, Disproofs, Algebraic Proofs, Boolean Algebras, Russell's Paradox and the Halting Problem. **DMA 6.1-6.4** **The Logic of Compound Statements**: Logical Form and Logical Equivalence, Conditional Statements, Valid and Invalid Arguments **DMA 2.1-2.5**	12
II	**Quantified Statements**: Predicates and Quantified Statements, Statements with Multiple Quantifiers, Arguments with Quantified Statements **DMA 3.1-3.4** **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. **DMA 4.1-4.8**	12

III	**Sequences, Mathematical Induction, and Recursion**: Sequences, Mathematical Induction, Strong Mathematical Induction and the Well-Ordering Principle for 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. **DMA 5.1-5.9** **Functions**: Functions Defined on General Sets, One-to-One and Onto, Inverse Functions, Composition of Functions, Cardinality with Applications to Computability **DMA 7.1-7.3**	**12**
IV	**Relations**: Relations on Sets, Reflexivity, Symmetry, and Transitivity, Equivalence Relations, Partial Order Relations **DMA 8.1-8.5** **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. **DMA 10.1-10.7**	**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. **DMA 9.1-9.9**	**12**

References :

DMA : Discrete Mathematics with Applications by Sussana S. Epp Cengage Learning 4th Edition 2010

DMS : Discrete Mathematics, Schaum's Outlines Series by Seymour Lipschutz, Marc Lipson Tata MCGraw Hill 2007

Additional References :

Discrete Mathematics and its Applications by Kenneth H. Rosen Tata MCGraw Hill

Discrete mathematical structures by B Kolman RC Busby, S Ross PHI

Discrete structures by Liu Tata MCGraw Hill

List of Practical: Write the programs for 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 for 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

Course Title : Communication Skills
Course Code : VESUSIT105

Objective:

To understand the communication process, apply the same skills in various situations and improve on interpersonal communication skills. Also to impart the correct practices of strategies of effective business writing

Learning Outcomes (LO):

On completion of the course, learner should be able to

LO1: understand and apply the knowledge of human communication, language process and foundations of communication for business purposes.
LO2: communicate their thoughts and ideas effectively in both written and verbal format
LO3: work with the group of people of an organization, to share their needs thoughts, plans, expertise, opinion and so on for successful business interactions
LO4: acquaint them to network-based collaborative technologies that make real time communication a reality and increase their productivity.
LO5: improve overall writing skills for professional needs, identify cultural differences of speaking and body language

Unit	Details	Lectures
I	**The Seven Cs of Effective Communication:** Completeness, Conciseness, Consideration, Concreteness, Clarity, Courtesy, Correctness **SS Pg 160-161** **Understanding Business Communication:** Nature and Scope of Communication, Non-verbal Communication, Cross-cultural communication, Technology-enabled Business Communication **MP Pg 3-126**	**12**
II	**Writing Business Messages and Documents:** Business writing, Business Correspondence, Instructions, Business Reports and Proposals, Career building and Resume writing. **MP Pg 133-253** **Developing Oral Communication Skills for Business:** Effective Listening, Business Presentations and Public Speaking, Conversations, Interviews **MP Pg 261-379**	**12**
III	**Developing Oral Communication Skills for Business:** Meetings and Conferences, Group Discussions and Team Presentations, Team Briefing, **MP Pg 384-445** **Understanding Specific Communication Needs:** Communication across Functional Areas **MP Pg 453-487**	**12**

IV	**Understanding Specific Communication Needs:** Corporate Communication, Persuasive Strategies in Business Communication, Ethics in Business Communication, Business Communication Aids **MP Pg 494-614**	**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, **MP Pg 296-324 AK Pg 356-372** Use of templates. Adding graphics to your presentation: Visual communication, Impress stage: use of font, color, layout, Importance of practice and performance. **AK Pg 344-348**	**12**

References:

MP: Business Communication Edited by Meenakshi Raman and Prakash Singh, 2nd Edition, Oxford University Press

AK: Professional Communication by Aruna Koneru, Tata McGraw Hill

MSR: Strategies for improving your business communication by Prof. M. S. Rao, Shroff publishers and distributors, 2016

RJ: Business Communication by Dr. Rishipal and Dr. Jyoti Sheoran, SPD, 2014

RC: Graphics for Learning: Proven Guidelines for Planning, Designing, and Evaluating Visuals in Training Materials by Ruth C. Clark and Chopeta Lyons, Pfeiffer, Wiley, 2011

LM: Basic Business Communication: Skills for Empowering the Internet Generation by Lesikar Raymond V and Marie E. Flatley, 10th Edition, Tata McGraw- Hill, 2005

RW: Nonverbal Communication: Notes on the Visual Perception of Human Relations by Ruesh, Jurgen and Weldon Kees, University of California Press, 1966

BCT: Business Communication Today by Bovee, Courtland L.; Thill, John V. , Pearson Education Ltd, 2015

NR: Communication Skills by Dr. Nageshwar Rao Dr. Rajendra P. Das, Himalaya Publishing House

SS: BUSINESS AND MANAGERIAL COMMUNICATION By SAILESH SENGUPTA, PHI

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 practicals, Microsoft Office, Open Office, Libre Office or any other software suite can be used.
8.	**Use of word processing tools for communication**
9.	**Use of spreadsheet tools for communication**
10.	**Use of presentation tools for communication**

SEMESTER II

Course Title : Object Oriented Programming
Course Code : VESUSIT201

Objective:

To execute and advance in constructing foundational skills for rudimentary object-oriented programming.

Course Outcome:

LO 1: Demonstrate proficiency in Object-Oriented Programming to demonstrate Usage of Classes, Objects, Constructors, and Destructors, and Overloading.

LO 2: Apply the concepts in object-oriented programming in terms of software reuse and managing complexity to solve real-world problems.

LO 3: Classify Inheritance with the understanding of Early and late Binding

LO 4: Understand the File I/O Operations, templates, and pointers

Unit	Details	Lectures
I	**Object Oriented Methodology**: Introduction, Advantages and Disadvantages of Procedure Oriented Languages, what is Object-Oriented? What is Object-Oriented Development? Benefits and Application of OOPS. EB : 1.3 - 1.4, 1.6- 1.8 **Principles of OOPS:** OOPS Paradigm, Basic Concepts of OOPS: Objects, Classes, Data Abstraction and Data Encapsulation, Inheritance, Polymorphism, Dynamic Binding. EB: 1.4 - 1.5	**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, and functions, Pointer to object, Array of pointer to object. Defining methods, Static members, and Comments. Memory allocation, Dynamic allocation: new and delete. EB: 5.1,5.3-5.4, 5.10- 5.18	**12**

	Constructors and Destructors: Introduction, Default Constructor, Parameterized Constructor, and examples, Destructors EB: 6.1 - 6.7, 6.11	
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. EB: 7.1 - 7.5, 7.7,7.8 **Virtual Functions:** Introduction and need, Pure Virtual Functions, Static Functions, this Pointer, abstract classes, virtual destructors. EB: 9.1 - 9.7	**12**
IV	**Program development using Inheritance:** Introduction, understanding inheritance, Advantages provided by inheritance, choosing the access specifier, Derived class declaration, derived class constructors, protected members, class hierarchies, multiple inheritance, multilevel inheritance, containership, hybrid inheritance. EB: 8.1 - 8.12 **Exception Handling:** Introduction, Exception Handling Mechanism, Concept of throw & catch with example EB:13.1 - 13.7	**12**
V	**Templates:** Introduction, Function Template and examples, Class Template and examples. EB: 12.1 -12.8 **Working with Files:** Introduction, File Operations, Various File Modes, File Pointer and their Manipulation EB: 11.1 -11.10	**12**

References :

EB: Object Oriented Programming with C++ by E. Balagurusamy, Tata McGraw Hill, 4th Edition.

Additional References:

Object Oriented Analysis and Design by Timothy Budd, TMH, 3rd Edition.

Mastering C++ by K R Venugopal, Rajkumar Buyya, T Ravishankar, Tata McGraw Hill, 2nd Edition.

C++ for beginners by B. M. Hirwani, SPD.

Effective Modern C++ by Scott Meyers,SPD.

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 repectively. Where getInfo() will be private method
b.	Design the class student containing getData() and displayData() as two of its methods which will be used for reading and displaying the student information respectively.Where getData() will be private method.
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.Where readNo() 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() and also 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.	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 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

Course title: Microprocessor Architecture

Course code:VESUSIT202

Objective: To understand and develop competence in architecture,programming and interfacing various hardware circuits to Microprocessors and advance processors

Learning Outcomes (LO):

LO1: To provide an understanding of microprocessor hardware and software. Can work with microprocessor based equipment, and can identify hardware from software faults
LO2: Describe the architecture and organization of Microprocessor along with the instruction set format.
LO3: Interfacing of memory & various I/O devices with 8085 microprocessor
LO 4: Analyze the data transfer information through serial parallel ports.
LO5: Develop assembly language programs using programming tools.

Unit	Details	Lectures
I	**Microprocessor, microcomputers, and Assembly Language:** Microprocessor,Microprocessor InstructionSet and Computer Languages, From Large Computers to Single-Chip Microcontrollers, Applications. **RG 1**.1-1.3 **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. **RG 2.1-2.5** **8085 Microprocessor Architecture and Memory Interface:** Introduction, 8085 Microprocessor unit, 8085-Based Microcomputer, Memory Interfacing, Interfacing the 8155 MemorySegment, Illustrative Example: Designing Memory for the MCTS Project,. CPU architecture,Processor organization RG 3.1-3.2,14.1-14.2 **8086 Microprocessor Architecture:** RG 18.3 DH chapter 3 and 4	**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. RG 11.1-11.5 **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. RG 4.1-4.5 **Introduction to 8085 Instructions:** Data Transfer Operations, Arithmetic Operations, Logic Operation,Branch Operation, Writing Assembly Languages Programs, Debugging a Program. RG 5.1-5.6	**12**
III	**Programming Techniques With Additional Instructions:** Programming Techniques: Looping, Counting andIndexing, Additional Data Transfer and 16-Bit ArithmeticInstructions, Arithmetic Instruction Related to Memory, Logic Operations: Rotate, Logics Operations: Compare, Dynamic Debugging. **RG 6**.1-6.6 **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. RG 7.1-7.5 **Stacks and Sub-Routines:** Stack, Subroutine, Restart, Conditional Call, Return Instructions, Advanced Subroutine concepts. RG 8.1-8.4	**12**
IV	**Code Conversion, BCD Arithmetic, and 16-Bit Data Operations:** BCD-to-Binary Conversion, Binary-to-BCD Conversion,Introduction to parallel processing concept Pipeline processing BCD Addition, BCD Subtraction, Introduction to Advanced Instructions and Applications, Multiplication, Subtraction With Carry. RG 9.1-9.9 **Software Development System and Assemblers:** Microprocessors-Based Software Development system, Operating System and Programming Tools, Assemblers and Cross-Assemblers, Writing Program Using Cross Assemblers. RG 10.1-10.2 **Interrupts:**	**12**

	The 8085 Interrupt, 8085 Vectored Interrupts, Restart as S/W Instructions, Additional I/O Concepts and processes. RG 12.1-12.4	
V	**The Pentium and Pentium Pro microprocessors:** Introduction, Special Pentium registers, Memory management, Pentium instructions, Pentium Pro microprocessor, Special Pentium Pro features. **DA,TS**-ch.1 **Core 2 and later Microprocessors:** Introduction, Pentium II software changes, Pentium IV and Core 2, i3, i5 and i7. SD- ch 7 **SUN SPARC Microprocessor:** Architecture, super scalar architecture SD- ch .5	**12**

References:

RG: Microprocessor architecture,programming and applications with the 8085/8080A by Ramesh Gaonkar. Wiley Eastern limited publication
DH: Microprocessors and Interfacing by Douglas v Hall. Tata Mcgraw-Hill Edition
SD- "Fundamentals of Computer Organization and Design,"by S. Dandamudi.Springer
SD: Guide to RISC Processor for Programmers and Engineers Springer; 2005th edition
DA,TS: Pentium processor system architecture By Don Anderson, Tom Shanley, MindShare Inc,

Additional References:

Computer System Architecture by M. Morris Mano. PHI
Structured Computer Organization. Andrew C. Tanenbaum PHI

List of Practical	
1.	**Perform the following Operations related to memory locations.**
a.	Store the data byte 32H into memory location 4000H.
b.	Exchange the contents of memory locations 2000H and 4000H
2.	**Simple assembly language programs.**
a.	Subtract the contents of memory location 4001H from the memory location 2000H and place the result in memory location 4002H.
b.	Subtract two 8-bit numbers.
c.	Add the 16-bit number in memory locations 4000H and 4001H to the 16-bit number in memory locations 4002H and 4003H. The most significant eight bits of the two numbers to be added are in memory locations 4001H and 4003H. Store the result in memory locations 4004H and 4005H with the most significant byte in memory location 4005H.

d.	Add the contents of memory locations 40001H and 4001H and place the result in the memory locations 4002Hand 4003H.
e.	Subtract the 16-bit number in memory locations 4002H and 4003H from the 16-bit number in memory locations 4000H and 4001H. The most significant eight bits of the two numbers are in memory locations 4001H and 4003H. Store the result in memory locations 4004H and 4005H with the most significant byte in memory location 4005H.
f.	Find the l's complement of the number stored at memory location 4400H and store the complemented number at memory location 4300H.
g.	Find the 2's complement of the number stored at memory location 4200H and store the complemented number at memory location 4300H.
3.	**Packing and unpacking operations.**
a.	Pack the two unpacked BCD numbers stored in memory locations 4200H and 4201H and store result in memory location 4300H. Assume the least significant digit is stored at 4200H.
b.	Two digit BCD number is stored in memory location 4200H. Unpack the BCD number and store the two digits in memory locations 4300H and 4301H such that memory location 4300H will have lower BCD digit.
4.	**Register Operations.**
a.	Write a program to shift an eight bit data four bits right. Assume that data is in register C.
b.	Program to shift a 16-bit data 1 bit left. Assume data is in the HL register pair
c.	Write a set of instructions to alter the contents of flag register in 8085.
d.	Write a program to count number of l's in the contents of D register and store the count in the B register.
5.	**Multiple memory locations.**
a.	Calculate the sum of series of numbers. The length of the series is in memory location 4200H and the series begins from memory location 4201H. a. Consider the sum to be 8 bit number. So, ignore carries. Store the sum at memory location 4300H. b. Consider the sum to be 16 bit number. Store the sum at memory locations 4300H and 4301H
b.	Multiply two 8-bit numbers stored in memory locations 2200H and 2201H by repetitive addition and store the result in memory locations 2300H and 2301H.
c.	Divide 16 bit number stored in memory locations 2200H and 2201H by the 8 bit number stored at memory location 2202H. Store the quotient in memory locations 2300H and 2301H and remainder in memory locations 2302H and 2303H.
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 2200H and the block itself begins in memory location 2201H. Store the number of negative elements in memory location

	2300H
e.	Find the largest number in a block of data. The length of the block is in memory location 2200H and the block itself starts from memory location 2201H. Store the maximum number in memory location 2300H. Assume that the numbers in the block are all 8 bit unsigned binary numbers.
6.	**Calculations with respect to memory locations.**
a.	Write a program to sort given 10 numbers from memory location 2200H in the ascending order.
b.	Calculate the sum of series of even numbers from the list of numbers. The length of the list is in memory location 2200H and the series itself begins from memory location 2201H. Assume the sum to be 8 bit number so you can ignore carries and store the sum at memory location 2Sample problem:
c.	Calculate the sum of series of odd numbers from the list of numbers. The length of the list is in memory location 2200H and the series itself begins from memory location 2201H. Assume the sum to be 16-bit. Store the sum at memory locations 2300H and 2301H.
d.	Find the square of the given numbers from memory location 6100H and store the result from memory location 7000H
e.	Search the given byte in the list of 50 numbers stored in the consecutive memory locations and store the address of memory location in the memory locations 2200H and 2201H. Assume byte is in the C register and starting address of the list is 2000H. If byte is not found store 00 at 2200H and 2201H
f.	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 6000H Write an assembly language program that adds these two numbers and stores the sum in the same format starting from memory location 6200H
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 array l, array2 and array3 are 2200H, 2300H and 2400H, respectively
7.	**Assembly programs on memory locations.**
a.	Write an assembly language program to separate even numbers from the given list of 50 numbers and store them in the another list starting from 2300H. Assume starting address of 50 number list is 2200H
b.	Write assembly language program with proper comments for the following: A block of data consisting of 256 bytes is stored in memory starting at 3000H. 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 2040H and the string itself begins in memory location 2041H. Place even parity in the most significant bit of each character.
d.	A list of 50 numbers is stored in memory, starting at 6000H. Find number of negative, zero and positive numbers from this list and store these results in memory locations 7000H, 7001H, and 7002H respectively
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.	Multiply the 8-bit unsigned number in memory location 2200H by the 8-bit unsigned number in memory location 2201H. Store the 8 least significant bits of the result in memory location 2300H and the 8 most significant bits in memory location 2301H.
d.	Divide the 16-bit unsigned number in memory locations 2200H and 2201H (most significant bits in 2201H) by the B-bit unsigned number in memory location 2300H store the quotient in memory location 2400H and remainder in 2401H
e.	DAA instruction is not present. Write a 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 40FFH to 40FFH. In case of any error, it is indicated by writing 01H at port 10
b.	Arrange an array of 8 bit unsigned no in descending order
c.	Transfer ten bytes of data from one memory to another memory block. Source memory block starts from memory location 2200H where as destination memory block starts from memory location 2300H
d.	Write a program to find the Square Root of an 8 bit binary number. The binary number is stored in memory location 4200H and store the square root in 4201H.
e.	Write a simple program to Split a HEX data into two nibbles and store it in memory
10.	**Operations on BCD numbers.**
a.	Add two 4 digit BCD numbers in HL and DE register pairs and store result in memory locations, 2300H and 2301H. Ignore carry after 16 bit.
b.	Subtract the BCD number stored in E register from the number stored in the D register
c.	Write an assembly language program to multiply 2 BCD numbers

Course title: Web Programming
Course code: VESUSIT203

Objective: To understand different concept for web site development and and develop a dynamic web site

Learning Outcomes (LO):

On successful completion of this course students will be able to:

LO1	Understand different uses of Internet
LO2	Effectively develop web pages
LO3	Incorporate JavaScript in a web page
LO4	Develop an Interactive Website

Unit no.	Details of topics	No of lectures
1	**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, web saver – 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.** **FW:**xix,pg.no 25 to 25,47 to 65,76 to 81,85 to 100,103 to 139	12 Lectures
2	**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. **FW:165-183,185 to 203** **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	12 Lectures

	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 on web page. **FW:Pg NO 205 to 229,231 to 249, 251 to 267,271 to 285**	
3	**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 TF:Chapter 1,2,3 **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, Timer Object and its Methods **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 TF:Chapter 4,5,7,8,9,10,11	15 Lectures
4	**PHP:** Why PHP and MySQL? Server-side scripting, PHP syntax and variables, comments, types, control structures, branching, looping, termination, functions, passing information with PHP, GET, POST, formatting form variables, superglobal arrays, strings and string functions, regular expressions, arrays, number handling, basic PHP errors/problems ST:Chapter 1,2,3,4,5,6,7,8,9	
5	**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 andHTTP, E-Mail ST:Chapter 7,15,16,1722,24,37	

References:

FW:HTML5 Step by Step Faithe Wempen Microsoft Press 2011
TF:JavaScript 2.0: The Complete Reference Thomas Powell and Fritz Schneider Tata McGraw Hill 2 nd
ST:PHP 6 and MySQL Bible Steve Suehring,Tim Converse,Joyce Park
Wiley 2009

Additional references:

Web Design The Complete Reference ByThomas Powell Tata McGraw Hill
PHP Project for Beginners SharanamShah,Vaishali Shah SPD 2015
PHP 5.1 for Beginners Ivan Bayross Sharanam Shah,SPD 2013
Head First HTML 5 programming Eric Freeman O'Reilly 2013

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.	Write a PHP program to accept a number from the user and print whether it is prime 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 table named “Student” with following fields (sno, sname, percentage). Insert 3 records of your choice. Display the names of the students whose percentage is between 35 to 75 in a tabular format.
c.	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.

Course title: **Numerical and Statistical Methods**
Course code: VESUSIT204

Objective: The learner will be exposed to the development, computation and application of optimal control algorithms required for scientific computing and data analysis.

Learning Outcomes (LO):

On successful completion of this course students will be able to:

LO1: Identify the role of errors , solve algebraic and transcendental equations
LO2: Apply various interpolations to find an unknown value from the set of given values related to a situation and find the solution of simultaneous algebraic equations using iterative methods.
LO3: Apply differentiation and integration using various rules like Trapezoidal Rule, Simpson's Rule, Euler's Method, Runge-Kutta Method.
LO4: Apply Linear Regression and Linear Programming Problems for any real life situation and compare the role of various distributions such as Uniform, Binomial, Poisson and Bernoulli.

Unit	Details	Lectures
I	**Approximations and Round-Off Errors:** Significant Figures, Accuracy and Precision, Error Definitions, Round-Off Errors **NME 3.1, 3.2, 3.3, 3.4** **Truncation Errors and the Taylor Series:** The Taylor Series, Error Propagation, Total Numerical Errors, Formulation Errors and Data Uncertainty **NME 4.1, 4.2, 4.3** **Solution of simultaneous algebraic equations (linear) using iterative methods:** Gauss-Jordan Method, Gauss-Seidel Method. **NME 9.7, 11.2**	12
II	**Solutions of Algebraic and Transcendental Equations:** The Bisection Method, The Newton-Raphson Method, The Regula-falsi method, The Secant Method. **IMNM 2.2, 2.3, 2.5, 2.7** **Interpolation:** Forward Difference, Backward Difference, Newton's Forward Difference Interpolation, Newton's Backward Difference Interpolation, Lagrange's Interpolation. **IMNM 3.3, 3.6, 3.9.1**	12

III	**Numerical differentiation and Integration:** Numerical differentiation, Numerical integration using Trapezoidal Rule, Simpson's 1/3rd and 3/8th rules. **IMNM 5.2, 5.4** **Numerical solution of 1st and 2nd order differential equations:** Taylor series, Euler's Method, Modified Euler's Method, Runge-Kutta Method for 1st and 2nd Order Differential Equations. **IMNM 7.2, 7.4, 7.5**	**12**
IV	**Least-Squares Regression:** Linear Regression, Polynomial Regression, Multiple Linear Regression, General Linear Least Squares, Nonlinear Regression **NME : 17** **Linear Programming:** Linear optimization problem, Formulation and Graphical solution, Basic solution and Feasible solution. **NME : 15**	**12**
V	**Random variables:** Discrete and Continuous random variables, Probability density function, Probability distribution of random variables, Expected value, Variance. **FMS 5.1, 5.2, 5.3, 5.4** **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. **FMS 7.1, 7.2, 7.3, 8.2**	**12**

Reference :

IMNM : Introductory Methods of Numerical Methods by S. S. Sastry PHI Vol – 2
FMS : Fundamentals of Mathematical Statistics by S. C. Gupta, V. K.Kapoor
NME : Numerical Methods for Engineers by Steven C. Chapra, Raymond P. Canale Tata McGraw Hill 6th Edition 2010

Additional References :

Numerical Analysis by Richard L. Burden, J. Douglas Faires Cengage Learning 9th Edition 2011

Elements of Applied Mathematics by P.N.Wartikar and J.N.Wartikar A. V. Griha, Pune Volume 1 and 2

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 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.	Programing to obtain derivatives numerically.
6.	**Numerical Integration**
a.	Program for numerical integration using Trapezoidal rule.
b.	Program for numerical integration using Simpson’s 1/3rd rule.
c.	Program for numerical integration using Simpson’s 3/8th rule.
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 2nd order and 4th 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.

Course title: **Green Computing**

Course code: VESUSIT205

Objective:

To be empowered to reduce the energy use, waste and other environmental impacts of IT systems and make green IT an integral part of organizational culture and planning.

Learning Outcomes (LO):

On completion of this course, learners should be able to
LO1: create awareness and promote green initiatives in the community
LO2: acquire knowledge about power consumption issues, ethical IT equipment disposal etc
LO3: understand the concept of 3 R's(Reduce, Reuse and Recycle)
LO4: implement initiatives and standards to build the greenest IT Department
LO5: generate and present the project report

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. **TAR Pg 3-18** **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. **TAR Pg 20-42**	**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. **TAR Pg 45-62** **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,	**12**

	Datacentre Design, Centralized Control, Design for Your Needs, Put Everything Together. **TAR Pg 63-77**	
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. **TAR Pg 81-101** **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. **TAR Pg 103-126**	**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 **TAR Pg 127-149** **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 **TAR Pg 151-170**	**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. **TAR Pg 255-275** **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, Analyze Data, Conduct Audits, Certifications, Benefits, Realities, Helpful Organizations. **TAR Pg 277-290**	**12**

References:

TAR: Green IT by Toby Velte, Anthony Velte, Robert Elsenpeter, 2008, McGraw Hill

AMM: Green Data Center: Steps for the Journey by Alvin Galea, Michael Schaefer, Mike Ebbers, 2011, Shroff Publishers and Distributers

JH: Green Computing and Green IT Best Practice by Jason Harris, Emereo

BS: Green Computing Tools and Techniques for Saving Energy, Money and Resources by Bud E. Smith, 2014, CRC Press

Project and Viva Voce	
1.	**A project should be done based on the objectives of Green Computing. A report of minimum 50 pages should be prepared. The report should have a font size of 12, Times new roman and 1.5 line spacing. The headings should have font size 14. The report should be hard bound.**
2.	**The project can be done individually or a group of two students.**
3.	**The students will have to present the project during the examination.**
4.	**A certified copy of the project report is essential to appear for the examination.**

Modality of assessment

The performance of the learners shall be evaluated into two parts. The learner’s performance shall be assessed by Internal Assessment with 25% marks in the first part & by conducting the Semester End Examinations with 75% marks in the second part. Practical Examination will consist of End Sem examination.

Student will have to score 40% of marks in Internal assessment as well as End Sem examination to pass the course.

The allocation of marks for the Internal Assessment and Semester End Examinations are as shown below:-

Internal Assessment: It is defined as the assessment of the learners on the basis of internal evaluation as envisaged in the Credit & Choice based system by way of participation of learners in various academic and correlated activities in the given semester of the programme.

Semester End Assessment : It is defined as the assessment of the learners on the basis of Performance in the semester end Theory/ written/ Practical examination.

A. Theory - Internal assessment 25% 25 marks

Sr No	Evaluation type	Marks
1.	- Class Test (multiple choice questions / objective)	15
2.	- Mini Project/Case study or Assignment	05
3.	- Active participation in routine class activity - Overall conduct as a responsible student, with respect to good behaviour, leadership qualities, interpersonal skills etc.	05

B. Theory - External examination - 75% 75 marks

Semester End Theory Assessment

Duration - Each paper shall be of 2.5 hours duration.

1. Theory question paper pattern :-
   a. There shall be Five compulsory questions, one based on each unit
   b. Each question shall carry 15 marks
      i. Each question shall be subdivided into six sub questions a, b ,c,d,e,and f with 5 marks each out of which student will have to attempt any three

Question no.	Details	Marks
Q1.	**(Based on Unit 1) Attempt any three out of Six:**	**15**
Q2.	**(Based on Unit 2) Attempt any three out of Six:**	**15**
Q3.	**(Based on Unit 3) Attempt any three out of Six:**	**15**
Q4.	**(Based on Unit 4) Attempt any three out of Six:**	**15**
Q5.	**(Based on Unit 5) Attempt any three out of Six:**	**15**

C. Semester End Practical Assessment 50 marks

	Based on Each Subject	50 Marks
A	Practical Question 1	20
B	Practical Question 2	20
C	Viva	05
D	Journal	05
	OR	
A	Practical Question 1	40
B	Viva	05
C	Journal	05

PRACTICAL BOOK/JOURNAL

The students are required to present a duly certified journal for appearing at the practical examination, failing which they will not be allowed to appear for the examination.

In case of loss of Journal and/ or Report, a Lost Certificate should be obtained from Head/ Coordinator / In-charge of the department ; failing which the student will not be allowed to appear for the practical examination.

Overall Examination and Marks Distribution Pattern

SEMESTER I

Course	VESUSMB101	VESUSMB102	Grand Total
Theory	100	100	200
Practicals	50	50	100

SEMESTER II

Course	VESUSMB201	VESUSMB202	Grand Total
Theory	100	100	200
Practicals	50	50	100