
SCHEME OF TEACHING
DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING
FIFTH SEMESTER – B.E

Sl. No	Course Code	Course Title	Teaching Dept	Contact Hours / Week				Credits
				L	T	P	Total	
1	IS5C01	MicroProcessor	IS & E	3	0	0	3	3
2	IS5C02	Systems Software	IS & E	3	0	2	5	4
3	IS5C03	Data Communication	IS & E	4	0	0	4	4
4	IS5C04	Operating Systems	IS & E	4	0	0	4	4
5	IS5C05	Engg. Management and Entrepreneurship	IS & E	3	0	0	3	3
6	IS5M06	MOOC Elective The Joy of Computing using Python	IS & E	3	0	0	3	3
7	IS5L01	MicroProcessor Lab	IS & E	0	0	3	3	1.5
8	IS5L02	Operating Systems Lab	IS & E	0	0	3	3	1.5
9	MA5CL1	Applied Mathematics – II *	MATHS	3	0	0	3	3
TOTAL				20	0	8	28	24
				23	0	8	31	27

* For lateral Entry Students

MOOC ELECTIVE

Course Code	Course Title	(L:T:P) Credit
IS5M01	Compiler Design	(3-0-0) 3
IS5M02	Computer graphics	(3-0-0) 3
IS5M03	Data Mining	(3-0-0) 3
IS5M04	Advanced Graph Theory	(3-0-0) 3
IS5M05	Software Defined Networks	(3-0-0) 3
IS5M06	The Joy of Computing using Python	(3-0-0) 3

SCHEME OF TECHING
DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING
SIXTH SEMESTER – B.E

Sl. No	Course Code	Course Title	Teaching Dept	Contact Hours / Week				Credits
				L	T	P	Total	
1	IS6C01	Software Architecture	IS & E	3	0	0	3	3
2	IS6C02	Web Technologies	IS & E	3	0	0	3	3
3	IS6C03	Computer Networks	IS & E	3	0	0	3	3
4	IS6C04	Cryptography and Network Security	IS & E	3	0	2	5	4
5	IS6C05	Engg. Economics	IS & E	3	0	0	3	3
6	IS6E1x	Dept. Elective – 1	IS & E	3	0	0	3	3
7	IS6L01	Web Technologies Lab	IS & E	0	0	3	3	1.5
8	IS6L02	Computer Networks Lab	IS & E	0	0	3	3	1.5
9	IS6C06	Minor Project	IS & E	0	0	4	4	2
TOTAL				18	0	12	30	24

DEPT. ELECTIVE – I

Course Code	Course Title	(L:T:P) Credit
IS6E101	Data Science and Data Analytics	(2-0-2) 3
IS6E102	Cyber Security	(3-0-0) 3
IS6E103	Introduction To Machine Learning	(3-0-0) 3
IS6E104	Object Oriented Analysis & Design	(3-0-0) 3
IS6E105	UX & UI	(3-0-0) 3

V SEMESTER

MICROPROCESSOR (3:0:0)

Sub code : IS5C01

Hrs/week : 03

SEE Hrs : 03

CIE : 50 Marks

SEE : 50% Marks

Max. Marks: 100

Course Outcomes:

On Successful completion of the course, the students will be able to:

1. Explain the internal architecture and assembler directives of 8086.
2. Illustrate the working of 8086 instructions set.
3. Describe macros, procedures and digital Interfacing using 8255 Programmable Parallel Ports.
4. Illustrate the basic concepts of shared memory and multi-threaded parallelism with programs.
5. Apply OpenMP constructs and clauses to tackle programming problems.

MODULE 1: Min mode and Max mode?

7 Hrs

An overview of Microprocessor: 8086 Internal Architecture, Addressing modes of 8086.

Assembly Directives: Assume, DB, DD, DQ, DT, DW, END, ENDP, ENDS, EQU, EVEN, OFFSET, PROC, PTR, Segment. .org -> origin MEMORY MODELS

Self Learning Exercise: Constructing the machine codes for 8086 Instructions.

MODULE 2:

8 Hrs

8086 Instruction Descriptions:

AAA, AAD, AAM, AAS, ADC, AND, CBW, CLC, CLD, CLI, CMP, CMPS, CWD, DAA, DAS, DEC, DIV, IN, INC, INT, IRET, JE, JMP(Unconditional), LAHF, LDS, LEA, LES, LODS, different LOOP instructions, MOV, MOVS, MUL, NEG, NOP, NOT, OR, OUT, POP, POPF, PUSH, PUSHF, different Rotate instructions, different Repeat instructions, SHAF, SAL, SAR, SBB, SUB, SCAS, STC, STD, STI, STOS, SUB, XCHG, XLAT, XOR.

Self Learning Exercise: Different Jump instructions (Conditional).

MODULE 3:

8 Hrs

Procedures and Macros: 8086 CALL and RET instructions, passing parameters to and from procedures, defining and calling MACRO, PROCEDURE versus MACROS.

Digital Interfacing: Programmable Parallel Port (8255) internal block diagram and system connection, operational mode and initialization, constructing and sending 8255 control word for Interfacing Microprocessor 8086 with I/O devices.

Self Learning Exercise: Interrupts, Types of Interrupts and applications.

Hardware programs

MODULE 4:**8 Hrs****OpenMP API for multi-threaded, shared memory parallelism.**

Introduction to shared memory and multi-threaded parallelism. Fork-join model used in OpenMP. Components of OpenMP- Compiler Directives. Using OpenMP Flag in GNU GCC compiler to build parallel programs, examples.

Self Learning Exercise: Runtime Library Routines and Environment Variables.

MODULE 5:**8Hrs**

OpenMP Programming

Clauses to control Parallel and Work-Sharing Constructs: Shared Clause, Private Clause, Default Clause, Nowait Clause, Lastprivate Clause, Firstprivate Clause.

OpenMP Synchronization Constructs: Barrier Construct, Ordered Construct, Critical Construct, Atomic Construct, Locks, Master Construct.

Self Learning Exercise: Matrix Times Vector Operation, Comparison of sequential and OpenMP based parallel programs.

TEXT BOOKS:

1. Microprocessors and Interfacing, Douglas V. Hall, 3rd Edition, TMH, 2012.
(Note: For Modules - 1, 2, 3 refer Text Book 1)
2. Using OpenMP: Portable Shared Memory Parallel Programming, Barbara Chapman, Gabriele Jost, Ruud van der Pas, The MIT Press Cambridge, Massachusetts London, England, 2008.
(Note: For Modules – 4, 5 refer Text Book 2)

REFERENCE BOOKS:

1. The Intel Microprocessors, Barry B. Brey, Pearson/PHI, 8th Edition, 2014.
2. The Intel Microprocessor Family: Hardware and Software Principles and Applications, James L. Antonakos, Thomson, 1st Edition, 2011.
3. Microprocessor 8086: Architecture, Programming and Interfacing, by Sunil Mathur, PHI, 2010.
4. Parallel Programming , Techniques and Application Using Networked Workstation and Parallel Computers, Michael Allen, Pearson, 2nd Edition, 2011.

BOOKS:

1. <https://www.wiziq.com/tutorials/microprocessors>
2. <http://www.openmp.org/resources/>
3. <https://www.slideshare.net/DhanashreePrasad/openmp-tutorial-for-beginners>

MOOCs:

1. <http://www.microprocessor.org/resources/free-online-courses>
2. https://onlinecourses.nptel.ac.in/noc17_cs39/preview

CO – PO MAPPING:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	M	S										
CO 2	S	S	S		W							
CO 3	M	S	M									
CO 4	M		M	M	S							
CO 5	M		M	M	S							

CO – PSO MAPPING:

	PSO1	PSO2	PSO3
CO 1	S	M	
CO 2	W	M	
CO 3	M	M	W
CO 4	M	S	W
CO 5	M	S	

SYSTEMS SOFTWARE (3:0:2)**Sub code : IS5C02****Hrs/week : 05****SEE Hrs : 03****CIE : 50 Marks****SEE : 50% Marks****Max. Marks: 100****Course Outcomes:****On Successful completion of the course, the students will be able to:**

1. Illustrate the relationship between system software and machine architecture.
2. Analyze the basics of assembler features.
3. Describe the working of loaders and linkers.
4. Utilize lex and yacc tools for implementing lexer and parser.

MODULE 1:**8 Hrs****Introduction:** System Software and Machine Architecture

Machine Architecture: The Simplified Instructional Computer (SIC) – SIC Machine Architecture. SIC/XE Machine Architecture. SIC Programming.

Self Learning Exercise: CISC & RISC**MODULE 2:****9 Hrs****Assemblers – 1:** Basic Assembler Function – A Simple SIC Assembler, Assembler Algorithm and Data Structures. Machine Dependent Assembler Features – Instruction Formats & Addressing Modes, Program Relocation**Self Learning Exercise:** MASM Assembler.**MODULE 3:****7 Hrs****Assemblers – 2:** Machine Independent Assembler Features – Literals, Symbol Definition Statements, Expression, Program Blocks, Control Sections and Programming Linking. Assembler Design Operations: One-Pass Assembler, Multi-Pass Assembler**Self Learning Exercise:** SPARC Assembler.**MODULE 4:****8 Hrs**Loaders and Linkers: Basic Loader Functions – Design of an Absolute Loader, A Simple Bootstrap Loader. Machine Dependent Loader Features: Relocation, **Program Linking**, Algorithm and Data Structures for a Linking Loader. Machine Independent Loader Features: Automatic Library search, Loader options, Loader Design options: Linkage editors**Self Learning Exercise:** Dynamic Linking.

MODULE 5:**7 Hrs**

Lex and Yacc –The Simplest Lex Program, Grammars, Parser-Lexer Communication, A YACC Parser, The Rules Section, Running LEX and YACC, Using LEX - Regular Expression, Examples of Regular Expressions

Using YACC – Grammars, Recursive Rules, Shift/Reduce Parsing, What YACC Cannot Parse, A YACC Parser - The Definition Section, The Rules Section, The LEXER, Compiling and Running a Simple Parser, Arithmetic Expressions and Ambiguity.

Self Learning Exercise: LEX vs. Hand- Written Lexers, Yacc- Symbol Tables

List of Sample Programs for Practice

1. Write a lex file to count the number of lines, words, and characters in the input.
2. Write a lex program to eliminate comment lines in a C program and copy the resulting program into a separate file.
3. Write a lex file to count the number of numbers appearing in the input. Count the number of integers (without a decimal) separately from the number of floating point numbers (with a decimal, and at least one digit on either side of the decimal).
4. Write a lex program to implement the find and replace operations in the given file.
5. Write a lex program to validate IP address.
6. Write a Lex program to accept input as C program and do error detection and correction. Check for un-terminated string constant in input C program; that is, a string constant beginning with double quotes and extended for more than one line. Intimate the error line numbers and the corrective actions to user.
7. Write a Yacc program that tells whether its input is a palindrome.
8. Write the lex file and the yacc grammar for an expression calculator. You need to deal with only '+' and '-' for integer inputs.
9. Enhance the above by adding support for enclosing expressions in parentheses.
10. Enhance the above by adding support for '*' and '/'.
11. Add support for simple decimal representation floating point numbers in your calculator. \
12. Add support for identifier based variables in your calculator, and assignment statements
13. Write a program to generate the symbol table for given program.
14. Write a program to recognize identifiers and reserved words

TEXT BOOKS:

1. System Software: An introduction to systems programming, Leland L Beck, Manjula D, 3rd Edition, Pearson Education Limited, 2016.
2. Lex and Yacc - John.R.Levine, Mason and Doug Brown, 2nd Edition, O'Reilly, Shroff Publishers & Distributors, 2013.

REFERENCE BOOK:

1. System Programming and Operating Systems, D M Dhamdhere, TATA McGraw Hill, 2nd Edition, Reprint 2015.
2. Systems Programming, Srimanta Pal . Oxford Higher Education, 2016.
3. Systems programming for small computer, by H Daniel Marcellus, Prentice-Hall, December 2016

EBOOK:

1. http://ebooks.lpude.in/computer_application/mca/term_4/DCAP507_SYSTEM_SOFTWARE.pdf
2. https://books.google.co.in/books?id=3hwi_zX8S_AC&pg=PA35&source=gbv_toc_r&cad=4#v=onepage&q&f=false

MOOCs:

1. <https://www.coursera.org/lecture/build-a-computer/unit-6-1-assembly-languages-and-assemblers-l4EGm>
2. <http://nptel.vtu.ac.in/econtent/courses/CSE/06CS51/index.php>

CO – PO MAPPING:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	S											
CO 2	S	S	M									
CO 3	S	M										
CO 4	S	M	M									

CO – PSO MAPPING:

	PSO1	PSO2	PSO3
CO 1	M		
CO 2	S	M	
CO 3	M		
CO 4	M	M	

DATA COMMUNICATION (4:0:0)**Sub code : IS5C03****Hrs/week : 04****SEE Hrs : 03****CIE : 50 Marks****SEE : 50% Marks****Max. Marks: 100****Course Outcomes:****On Successful completion of the course, the students will be able to:**

1. Discuss the concepts of data communication and network models.
2. Outline Digital –to- Digital conversion and Analog to Digital conversion.
3. Describe different digital-to-analog conversion and analog-to-analog conversion techniques for transmission.
4. Explain error detection and correction techniques and calculate hamming distance.
5. Summarize the data link layer protocols and framing.
6. Discuss Access control , Channelization & IEEE Standards.

MODULE 1:**8 Hrs****Introduction:**

Data Communications, Networks, Network Types

Network Models:

Protocol layering, TCP/IP Protocol Suite, The OSI Model.

Self Learning Exercise: Addressing – Physical, Logical and Port addressing.**MODULE 2:****9 Hrs****Data, Signals and Digital Transmission:**

Analog and digital, Digital signals, Transmission impairment, Data rate limits, Performance, Digital-to-

Digital conversion, Analog-to- Digital conversion**Self Learning Exercise:** Transmission modes.**MODULE 3:****9 Hrs****Analog Transmission and Multiplexing-I:**

Digital–to–Analog conversion – ASK, FSK, PSK, Analog–to–Analog conversion – AM, FM, PM,

Multiplexing – FDM, TDM**Self Learning Exercise:** WDM**MODULE 4:****9 Hrs****Error Detection and Correction:**

Introduction to error detection correction, Block coding – Error detection, correction, hamming distance, Min Hamming distance, Linear block codes, Cyclic codes.

Self Learning Exercise: Checksum.**Polynomials??**

MODULE 5:**8 Hrs****Data Link Control:**

DLC Services: Framing; Flow and Error control, Data-Link Layer Protocols: Simple protocol, Stop-And-Wait protocol, Piggybacking, HDLC and Point-to-point Protocol: Services, Framing.

Self Learning Exercise: Transition Phases.

MODULE 6:**9 Hrs****Media Access Control (MAC), Wired LAN's:**

Random Access: CSMA, CSMA/CA, CSMA/CD, Controlled Access: Reservation, Polling, Token Passing, Channelization – FDMA, TDMA and Ethernet: Standard Ethernet

Self Learning Exercise: CDMA.

TEXT BOOK:

1. Data Communications and Networking, Behrouz A. Forouzan, 5th Edition, McGraw-Hill, 2014.

REFERENCE BOOKS:

1. Data and Computer Communication, William Stallings, 8th Edition.
2. Computer Networks: A Systems Approach, Larry L Peterson and Bruce S David, Elsevier, 4th Edition

EBOOK:

1. http://www.tutorialspoint.com/data_communication_computer_network/data_communication_computer_network_tutorial.pdf

MOOCs:

1. <http://www.nptelvideos.in/2012/11/data-communication.html>

CO – PO MAPPING:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	S	M			W							
CO 2	M											
CO 3	S											
CO 4		M		M								
CO 5		M			W							
CO 6	S											

CO – PSO MAPPING:

	PSO1	PSO2	PSO3
CO 1			
CO 2	M		
CO 3	M		
CO 4	W		
CO 5	M		
CO 6	M		

OPERATING SYSTEMS (4:0:0)

Sub code : IS5C04

Hrs/week : 04

SEE Hrs : 03

CIE : 50 Marks

SEE : 50% Marks

Max. Marks: 100

Course Outcomes:

On Successful completion of the course, the students will be able to:

1. Explain the concept of process and threads.
2. Analyze the different techniques used for interprocess communication and scheduling.
3. Outline the different mechanisms used in memory management.
4. Describe file system and I/O implementation.
5. Illustrate the concept of deadlock avoidance and deadlock prevention.
6. Explain the basics of Linux OS.

MODULE 1:

8 Hrs

Introduction, Processes and Threads:

What Is An Operating System: The Operating System as an Extended Machine, The Operating System as a Resource manager?

Processes: The Process Model, Process Creation, Process Termination, Process Hierarchies, Process States.

Threads: The Thread Model, Thread Usage, Implementing Threads in User Space, Implementing Threads in the Kernel, Hybrid Implementations

Self Learning Exercise: Modeling Multiprogramming, Scheduler Activations, Pop-Up Threads

MODULE 2:

9 Hrs

Interprocess Communication:

Race Conditions, Critical Regions, Mutual Exclusion with Busy Waiting, Sleep and Wakeup, Semaphores, Mutexes, Monitors, Message Passing.

Classical IPC Problems: The Dining Philosophers Problem, The Readers and Writers problem.

Scheduling: Introduction to Scheduling, Scheduling in Batch Systems, Scheduling in Interactive Systems, Scheduling in Real-Time Systems

Self Learning Exercise: Thread Scheduling

MODULE 3:

10 Hrs

Memory Management:

No Memory Abstraction, A Memory Abstraction: Address Spaces - The Notion of an Address Space, Swapping.

Virtual Memory: Paging, Page Tables, Speeding Up Paging

Page Replacement Algorithms: The Optimal Page Replacement Algorithm, The Not Recently Used Page Replacement Algorithm, The First-In, First-Out, The Second Chance Page Replacement

Algorithm, The Clock Page Replacement Algorithm, The Least Recently Used.

Design Issues For Paging Systems: Local versus Global Allocation Policies, Load Control, Page Size, Separate Instruction and Data Spaces, Shared Pages, Virtual Memory Interface.

Implementation Issues: Operating System Involvement with Paging, Page Fault Handling.

Segmentation: Implementation of Pure Segmentation

Self Learning Exercise: Segmentation with Paging.

MODULE 4:

8 Hrs

Input / Output:

I/O Software Layers: Interrupt Handlers, Device Drivers, Device-Independent I/O Software.

Clocks: Clock Software.

File Systems:

Directories: Single-Level Directory Systems, Hierarchical Directory Systems.

File System Implementation: File System Layout, Implementing Files, Implementing Directories, Shared Files.

Self Learning Exercise: Disk Space Management

MODULE 5:

9 Hrs

Deadlocks:

Introduction to Deadlocks: Conditions for Resource Deadlock.

Deadlock Detection And Recovery: Deadlock Detection with One Resource of Each Type, Deadlock Detection with Multiple Resource of Each Type, Recovery from Deadlock.

Deadlock Avoidance: Resource Trajectories, Safe and Unsafe States, The Banker's Algorithm for a Single Resource, The Banker's Algorithm for Multiple Resources.

Deadlock Prevention: Attacking the Mutual Exclusion Condition, Attacking the Hold and Wait Condition.

Self Learning Exercise: Attacking the No Preemption Condition, Attacking the Circular Wait Condition.

MODULE 6:

8 Hrs

Overview of Linux – Linux goals, Interface to Linux, The shell, Linux Utility Programs, Kernel structure

Processes in Linux – Fundamental concept, Process Management systems calls in Linux, Implementation of Processes and Threads in Linux, scheduling in Linux

Memory Management in Linux – Fundamental concept

Self Learning Exercise: Linux file system, file system calls in Linux

TEXT BOOK:

1. Modern Operating Systems, Andrew S Tanenbaum, 4th Edition, Prentice Hall, 2015.

REFERENCE BOOKS:

1. Operating Systems: A Concept Based Approach, D M Dhamdhare, 2nd Edition. Tata McGrawill
2. Operating System Principles, Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, 7th Edition, Wiley Publication
3. Operating Systems Internals and Design Principles, William Stallings, 7th Edition, Pearson Education Limited, 2016.

EBOOKS:

1. Operating Systems Study Guide by Tim Bower
2. Operating Systems and System Administration by Nigel Gunton

MOOCs:

1. <http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-828-operating-system-engineering/>
2. <http://web.stanford.edu/~ouster/cgi-bin/cs140-winter16/index.php>

CO – PO MAPPING:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	S											
CO 2	S	S	M									
CO 3	S	S	W									
CO 4	S	S	W									
CO 5	S	S	W									
CO 6	S	W	M									

CO – PSO MAPPING:

	PSO1	PSO2	PSO3
CO 1	M		
CO 2	S	M	
CO 3	S	M	
CO 4	M	M	
CO 5	M	M	
CO 6	W	M	

ENGINEERING MANAGEMENT AND ENTREPRENEURSHIP (3:0:0)

Sub code : IS5C05**Hrs/week : 03****SEE Hrs : 03****CIE : 50 Marks****SEE : 50% Marks****Max. Marks: 100****Course Outcomes:****On Successful completion of the course, the students will be able to:**

1. Explain the development of management thought and Concept of Entrepreneurs.
2. Evaluate the human behavior concepts and HRM.
3. Illustrate financial statements and concepts of Marketing.
4. Apply the project management tools to manage projects.
5. Describe the concepts of Quality and Industrial management.

MODULE 1:**8 Hrs**

Entrepreneurship –Introduction, Management & Administration, Types of ownership and Organization structures. Concept of Entrepreneur, kind of Entrepreneurs, Entrepreneurship development and Govt. support in India. Role of Entrepreneurs in Economic Development

Self Learning Exercise: Barriers to Entrepreneurship

MODULE 2:**8 Hrs**

Human Resource Management: Functions of HRM, Recruitment and Selection, Interviewing Candidates. Human Resource Development, Training and Development, Performance Appraisal and Employee Compensation

Organizational Behaviour: Motivation, Content Theories: Maslow and Herzberg, Stress and Conflict, Management by Objectives, Job Enrichment, Job rotation

Self Learning Exercise: Individual and Group Behaviour. Negotiation

MODULE 3:**8 Hrs**

Marketing Management: Introduction, 5 Ps of Marketing, product life cycle, market Strategy U

Financial Management: Introduction, Types of Finance, Balance Sheet and Profit and Loss account statement, working capital, International Finance

Self Learning Exercise: Types of Taxes

MODULE 4:**7 Hrs**

Project Management: Project/Program/Portfolio Management, Phases in Project life cycle, Top Down and Bottoms up Estimation, WBS , Stake Holder Management. Identification of new Ideas, Evaluation of Alternatives

Self Learning Exercise: Make in India

MODULE 5:**8 Hrs**

Quality Management: Introduction, Contribution of Quality Gurus- Edward Deming (PDCA cycle), Joseph Juran (Quality trilogy), Quality Tools.

Industrial Management: Innovation in science, technology and industry: IOT, Big Data and Analytics. Lean and Six Sigma, 5S Techniques, Energy Management.

Self Learning Exercise: TQM, Industry 4.0 and Digital Manufacturing

TEXT BOOKS:

1. Management and Entrepreneurship - (Sixth Edition) K R Phaneesh, Sudha Publication, Year 2013.

REFERENCE BOOKS:

1. Quality control and Total quality Management, (6th Edition) Tata McGraw Hill, Year 2006.
2. Organizational Behavior- Stephen P. Robbins, Pearson Education India, 2009
3. Engineering economics-R. Panneerselvam (2nd Edition), PHI Learning Pvt. Ltd., 2013
4. Marketing Management by Philip Kotler, Kevin Lane Keller, Pearson publication.
5. Project Planning, Analysis, Selection, Implementation and. Review- Prasanna Chandra, Tata McGraw Hill Publications, New Delhi, 2000
6. Energy Management Principles (2nd Edition), Craig B. Smith Kelly Parmenter
7. Industrial Management, D K Bhattacharyya Vikas Publishing
8. Financial Management- I.M. Pandey (9th Edition) “Financial Management”, Vikas publication, 2011

CO – PO MAPPING:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	M	S	M	M		W		M	M		M	M
CO 2	S	S	M			M		M	M	W		M
CO 3	M	M				M						W
CO 4	S	S					M					M
CO 5	W	M				M		M		M		M

CO – PSO MAPPING:

	PSO1	PSO2	PSO3
CO 1		W	M
CO 2	W	M	M
CO 3		M	W
CO 4	W	S	M
CO 5		M	M

MOOC ELECTIVE (3:0:0)
THE JOY OF COMPUTING USING PYTHON
(Syllabus as NPTEL Portal)

Sub code : IS5M06
Hrs/week : 03
SEE Hrs : 03

CIE : 50 Marks
SEE : 50% Marks
Max. Marks: 100

Course Outcomes:

On Successful completion of the course, the students will be able to:

1. Illustrate basics of programming and python.
2. Use List types in Python for data storage and processing.
3. Use Dictionary in python for data storage and processing
4. Demonstrate the feature of Tuples and advanced function.
5. Explain the usage of special libraries and NetworkX in python

MODULE 1:

7 Hrs

Introduction to Programming: Why Programming? Programming for Everybody, Any Prerequisites? Where to start? Why do we have so many languages? How to go about programming? Why to learn programming? What is programming? How to give instructions? Introduction to Scratch, Introduction to Loops, More about Loops, Solution to Looping Problem.

Installation of Anaconda: Introduction to Anaconda, Installation of Anaconda, Introduction to Spyder IDE, Printing statements in Python, Understanding Variables in Python, Executing a sequence of instructions in the Console, Writing your First Program, Taking inputs from the user, Discount Calculation, Motivation to if condition, A reminder on how to deal with numbers, Understanding if condition's working, Realizing the importance of syntax and indentation, Introductions to loops, Loops: Sum of numbers Multiplication Tables, Introduction to While Loop.

MODULE 2:

8 Hrs

Lists: Introduction, Manipulation, Operations, Slicing, Loops and Conditionals: Fizzbuzz, Crowd Computing - Just estimate, Permutations - Jumbled Words, Theory of Evolution
Magic Square: Hit and Trial, Double Game - Spot the similarity, Birthday Paradox - Find your twin, Guess the Movie Name

MODULE 3:

8 Hrs

Introduction to Dictionaries, Speech to Text: No need to write , Monte Hall : 3 doors and a twist, Rock, Paper and Scissor : Cheating not allowed !!
Substitution Cipher -The science of secrecy, Tic Tac Toe - Down the memory Lane, Recursion.

MODULE 4:**8 Hrs**

Snakes and Ladders - Not on the Board, Spiral Traversing - Let's Animate, GPS - Track the route. Tuples- Python Data Structure, Lottery Simulation - Profit or Loss, Image Processing - Enhance your images, Anagrams, Face book Sentiment Analysis, Natural Language Processing - Author Stylometry.

MODULE 5:**8 Hrs**

Introduction to Networkx , Six Degrees of Separation : Meet your favourites, Area Calculation - Don't Measure. FLAMES, Data Compression , Browser Automation Watsapp using Python , Fun with Calendar, Page Rank - How does Google Work ?, Collatz Conjecture

WEB LINKS:

1. MOOC: <https://swayam.gov.in>
2. Official Python Documentation: <https://docs.python.org>
3. Full Stack Python: <https://www.fullstackpython.com/>.

CO – PO MAPPING:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	M	W	M	M	M							
CO 2	M	W	M	M	M							
CO 3	M	W	S	S	S							
CO 4	M	W	S	S	S							
CO 5	M	W	S	S	S							

CO – PSO MAPPING:

	PSO1	PSO2	PSO3
CO 1	W	M	
CO 2	W	M	
CO 3	W	M	
CO 4	W	M	
CO 5	W	M	

MICROPROCESSOR LAB (0:0:3)**Sub code : IS5L01****Hrs/week : 03****SET Hrs : 03****CIE : 25 Marks****SET : 25 Marks****Max. Marks: 50****Course Outcomes:****On Successful completion of the course, the students will be able to:**

1. Demonstrate programs using different 8086 instructions and OpenMP API's.
2. Demonstrate interfacing of I/O device with 8086 microprocessor.

PART – A**(8086 Assembly Language Programming)**

1. Write separate ALPs to add, to subtract and to find an average of two numbers.
2. Write an ALP to check given number is positive or negative.
3. Write an ALP to find the largest of N numbers.
4. Write an ALP to find whether the given string is palindrome or not.
5. Write an ALP to perform binary search and display the output on the monitor.

(OpenMP Programming)

6. Write a program to print Hello World from multiple threads using OpenMP.
7. Write a program to generate Fibonacci series using OpenMP.
8. Write a program for Matrix multiplication using OPENMP.

PART – B

1. Read status of eight input bits from the Logic Controller Interface and display FF if it is even parity bits otherwise displays 00. Also display number of 1,s in the input data.
2. Perform the following functions using the Logic Controller Interface.
 - a. BCD up-down counter
 - b. Ring counter
 - c. Jonson counter
3. Display message FIRE and HELP alternately with flickering effects on a seven segment display interface for a suitable period of time.
4. Scan 3X8 Keypad for key closure and to store the code of the key pressed in a memory location or display it on the screen. Also display row and column of the key pressed.

TEXT BOOK:

1. Microprocessors and Interfacing, Douglas V. Hall, 3rd Edition, TMH, 2012.
2. Using OpenMP: Portable Shared Memory Parallel Programming, Barbara Chapman, Gabriele Jost, Ruud van der Pas, The MIT Press Cambridge, Massachusetts London, England, 2008.

REFERENCE BOOKS:

1. The Intel Microprocessors, Barry B. Brey, Pearson/PHI, 8th Edition, 2014.
2. The Intel Microprocessor Family: Hardware and Software Principles and Applications, James L. Antonakos, Thomson, 1st Edition, 2011.
3. Microprocessor 8086: Architecture, Programming and Interfacing, by Sunil Mathur, PHI, 2010.
4. Parallel Programming, Techniques and Application Using Networked Workstation and Parallel Computers, Michael Allen, Pearson, 2nd Edition, 2011.

EBOOKS:

1. <https://www.wiziq.com/tutorials/microprocessors>
2. <http://www.openmp.org/resources/>
3. <https://www.slideshare.net/DhanashreePrasad/openmp-tutorial-for-beginners>

MOOCs:

1. <http://www.microprocessor.org/resources/free-online-courses>
https://onlinecourses.nptel.ac.in/noc17_cs39/preview

CO – PO MAPPING:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	S	S	M		W							
CO 2	M	S	M		W							

CO – PSO MAPPING:

	PSO1	PSO2	PSO3
CO 1	S	S	
CO 2	S	S	

OPERATING SYSTEMS LAB (0:0:3)

Sub code : IS5L02
Hrs/week : 03
SET Hrs : 03

CIE : 25 Marks
SET : 25 Marks
Max. Marks: 50

Course Outcomes:

On Successful completion of the course, the students will be able to:

1. Demonstrate process management concepts.
2. Demonstrate memory management algorithms.

List of Programs:

1. Write a 'c' program to print the command line arguments. (Only for Practice)
2. Write a 'c' program to print the process identification number and its parent identification number.
3. Write a 'c' program to make parent wait to get response from more than one child.
4. Simulate the following CPU Scheduling Algorithms
 - a) FCFS
 - b) SJF
 - c) Priority
 - d) Round Robin
5. Simulate MVT (Multiprogramming with a Variable number of Tasks) and MFT (Multiprogramming with a Fixed number of Tasks)
6. Simulate page replacement algorithm for FIFO and LRU
7. Write a C program to simulate the following contiguous memory allocation techniques
 - a) Worst-fit
 - b) Best-fit
 - c) First-fit
8. Write a C program to simulate producer-consumer problem using semaphores.
9. Simulate paging techniques of Memory Management
10. Simulate to solve Dining philosopher problem
11. Write a C program to simulate Bankers algorithm for the purpose of deadlock avoidance.

TEXT BOOK:

1. **Modern Operating Systems**, Andrew S Tanenbaum, 4th Edition, Prentice Hall, 2014.

REFERENCE BOOK:

1. **Operating Systems: A Concept Based Approach**, D M Dhamdhere, 2nd Edition. Tata McGraw ill
2. **Operating System Principles**, Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, 7th Edition, Wiley Publicatin
- Operating system – Design and Implementation**, Tanenbaum A S, Woodhull Albert S, 3rd Edition, Pearson India Education, 2009.

EBOOKS:

1. Operating Systems Study Guide by Tim Bower
2. Operating Systems and System Administration by Nigel Gunton

MOOCs:

1. <http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-828-operating-system-engineering/>
2. <http://web.stanford.edu/~ouster/cgi-bin/cs140-winter16/index.php>

CO – PO MAPPING:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	M	S	M	W	W							
CO 2	S	M	S	M								

CO – PSO MAPPING:

	PSO1	PSO2	PSO3
CO 1	S	M	W
CO 2	S	M	

APPLIED MATHEMATICS – II (3:0:0)**(For Lateral Entry)****Sub code : MA5CL1****Hrs/week : 03****SEE Hrs : 03****CIE : 50 Marks****SEE : 50% Marks****Max. Marks: 100****Course Outcomes:****On Successful completion of the course, the students will be able to:**

VI SEMESTER

SOFTWARE ARCHITECTURE (3:0:0)

Sub code : IS6C01**Hrs/week : 03****SEE Hrs : 03****CIE : 50 Marks****SEE : 50% Marks****Max. Marks: 100****Course Outcomes:****On Successful completion of the course, the students will be able to:**

1. Recognize the need for software architecture processes and business cycle.
2. Explain different architectural styles used in software development
3. Describe the various system architecture quality attributes.
4. Discuss different architectural patterns. (Module 4 & 5)

MODULE 1:**7 Hrs****Introduction**

The Architecture Business Cycle: Where do architectures come from? Software processes and the architecture business cycle; what makes a “good” architecture? What software architecture is and what it is not; other points of view; Architectural patterns, reference models and reference architectures; Importance of software architecture;

Self Learning Exercise: Architectural structures and views.

MODULE 2:**8 Hrs****Architectural Styles and Case Studies**

Architectural styles; Pipes and filters; Data abstraction and object-oriented organization; Event-based, implicit invocation; Layered systems; Repositories; Interpreters; Process control; Other familiar architectures; Heterogeneous architectures. Case Studies: Keyword in Context; Instrumentation software; Mobile robotics; Cruise control;

Self Learning Exercise: Three vignettes in mixed style.

MODULE 3:**7 Hrs****Quality**

Functionality and architecture; Architecture and quality attributes; System quality attributes; Quality attribute scenarios in practice; Other system quality attributes; Business qualities; Architecture qualities. Achieving Quality: Introducing tactics; Availability tactics; Modifiability tactics; Performance tactics; Security tactics; Testability tactics; Usability tactics; Architectural patterns and styles.

Self Learning Exercise: Relationship of tactics to architectural patterns

MODULE 4:**10 Hrs****Architectural Patterns – 1**

Introduction; from mud to structure: Layers, Pipes and Filters, Blackboard. Distributed Systems: Broker.

Self Learning Exercise: Presentation-Abstraction-Control.

MODULE 5:**7 Hrs****Architectural Patterns – 2**

Interactive Systems: MVC, Adaptable Systems: Microkernel;

Self Learning Exercise: Reflection

TEXT BOOKS:

1. Len Bass, Paul Clements, Rick Kazman: Software Architecture in Practice, 2nd Edition 2003
2. Pattern Oriented Software Architecture: A System of Patterns, Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, Michael Stal, Volume 1, 1996. (Module 4)

REFERENCE BOOK:

1. Software Architecture: Foundations, Theory, and Practice 1st Edition, 2009 by R. N. Taylor and N. Medvidovic.

EBOOK:

1. Software Architecture in Practice 2nd Edition, Kindle Edition

MOOC:

1. <http://www.nptel.ac.in/syllabus/106104027/>

CO – PO MAPPING:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	S											
CO 2	W	S										
CO 3		S										
CO 4	W	S	M									

CO – PSO MAPPING:

	PSO1	PSO2	PSO3
CO 1	S	M	
CO 2		S	
CO 3		S	
CO 4	S	S	

WEB TECHNOLOGIES (3:0:0)**Sub code : IS6C02****Hrs/week : 03****SEE Hrs : 03****CIE : 50 Marks****SEE : 50% Marks****Max. Marks: 100****Course Outcomes:****On Successful completion of the course, the students will be able to:**

1. Create Web pages using HTML, XHTML & CSS.
2. Prepare dynamic web pages using JavaScript.
3. Use Document Type Definition in an XML document.
4. Use Angular in web application.
5. Develop Server Side Programming

MODULE 1:**8 Hrs**

Fundamentals of Web: Internet, WWW, Web Browsers and Web Servers, URLs, MIME, HTTP, Security. Introduction to XHTML: Basic syntax, Standard HTML Document Structure, Basic text markup, Images, Hypertext Links, Lists. Cascading Style Sheets: Introduction, Levels of style sheets, Style specification formats, Selector forms, Property value forms, Font, List properties, Span and div tag.

Self Learning Exercise: Tables**MODULE 2:****8 Hrs**

Basics of JavaScript: Overview of JavaScript, Object orientation and JavaScript, Syntactic characteristics, Primitives, operations, and expressions, Screen output and keyboard input, Control statements, Object creation and modification, Arrays, Functions, Constructors, Pattern matching using regular expressions, Errors in scripts, Examples. JavaScript and XHTML Documents: The JavaScript execution environment, The Document Object Model, Element access in JavaScript, Events and event handling, handling events from the Body elements, Button elements, Text box and Password elements, The DOM 2 event model, The canvas Element

Self Learning Exercise: The navigator object, DOM tree traversal and modification.**MODULE 3:****7 Hrs**

Introduction to XML: Introduction, Syntax, Document structure, Document type definitions, Namespaces, XML schemas, displaying raw XML documents, Displaying XML documents with CSS, XSLT style sheets.

Self Learning Exercise: XML processors, Web services.

MODULE 4:**8 Hrs**

Introduction to Typescript: Learning different types, Understanding interfaces and functions. Implementing classes and modules

Introduction to Angular: Understanding Angular, adding angular to your environment, using angular CLI, Creating basic angular application, Angular component: component configuration, Building a template, using external templates

Self Learning Exercise: Injecting directives

MODULE 5:**8 Hrs**

Introduction to PHP: Origins and uses of PHP, Overview of PHP, General syntactic characteristics, Primitives, operations and expressions, Output, Control statements, Arrays, Functions, Pattern matching, Form handling, Files, Session tracking, Database access with PHP and MySQL

Self Learning Exercise: Cookies

TEXT BOOKS:

1. Programming the World Wide Web, Robert W Sebesta, 8th Edition, Addison Wesley, 2014
2. Learning Angular: A Hands-on Guide to Angular 2 and Angular 4, Brad Dayley, Brendan Dayley, Caleb Dayley, 2nd Edition, Addison-Wesley Professional, 2017

REFERENCE BOOKS:

1. Internet & World Wide Web How to Program, M. Deitel, P.J. Deitel, A. B. Goldberg, 4th Edition.
2. Open Source Web Development with LAMP: Using Linux, Apache, MySQL, Perl, and PHP, James Lee & Brent Ware, Addison Wesley; 1st edition, 2002. (MODULE – 5)
3. Web Programming Building Internet Applications, Chris Bates, 3rd Edition.
4. The web Warrior Guide to Web Programming, Cengage Learning, Xue Bai.

EBOOKS:

1. Introduction to Web Applications Development by Carles Mateu Publisher: Free Technology Academy - Fundació per a la Universitat Oberta de Catalunya (February, 2010)
2. Learning Web Design by Jennifer niederst robbins.O'REILLY publications,3rd Edition.

MOOCs:

1. www.w3schools.com
2. php.net/manual/en/tutorial.php
3. https://developer.mozilla.org/en-US/Learn/Getting_started_with_the_web/JavaScript_basics

CO – PO MAPPING:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	M	M										
CO 2	S	S			S							
CO 3	W											
CO 4	W	S	S	M								
CO 5	S	S	M	W	M							

CO – PSO MAPPING:

	PSO1	PSO2	PSO3
CO 1	M	S	
CO 2		S	
CO 3		S	
CO 4		S	
CO 5		S	

COMPUTER NETWORKS (3:0:0)**Sub code : IS6C03****Hrs/week : 03****SEE Hrs : 03****CIE : 50 Marks****SEE : 50% Marks****Max. Marks: 100****Course Outcomes:****On Successful completion of the course, the students will be able to:**

1. Explain the concepts of IP address, Classes, IP datagrams.
2. Describe the mapping of logical to physical addresses and vice versa, and also the different protocols for allocating IP addresses.
3. Describe the forwarding and routing algorithms.
4. Discuss the protocols for process to process communication congestion control algorithms.
5. Discuss the importance of various application related services and protocols.

MODULE 1:**8 Hrs****Network Layer –I (*Logical Addressing, Internet Protocol*)**

Introduction, IPv4 Addresses: Address Space, Notations, Classful Addressing, Classless Addressing, Network Address Translation (NAT). IPv6 Addresses: Structure, Address Space. Internetworking: Need for Network Layer, Internet as a Datagram Network, Internet as a Connectionless Network. IPv4: Datagram, Fragmentation, Checksum, options, **IPv6: advantages ,packet format, comparison between IPv4 and IPv6 headers, transition from IPv4 to IPv6,** [Ipv6 external headers??](#)

Self Learning Exercise: Extension headers.**MODULE 2:****8 Hrs****Network Layer –II (*Address Mapping, Error Reporting and Multicasting*)**

Address Mapping: Mapping Logical to Physical Address: ARP, Mapping Physical to Logical Address: RARp, BOOTP, and DHCP. ICMP: Types of Messages, Message Format, Error Reporting messages, Query messages. IGMP: Group Management, IGMP Messages, Message Format, IGMP Operation,.

Self Learning Exercise: Physical Multicast Support .**MODULE 3:****7 Hrs****Network Layer –III (*Delivery, Forwarding and Routing*)**

Delivery: Direct Versus Indirect Delivery. Forwarding: Forwarding Techniques, Forwarding Process, **Routing Table**. Unicast Routing Protocols:, Optimization, Intra- and Interdomain Routing, RIP, Distance Vector Routing, Link State Routing,.

Self Learning Exercise: OSPF

MODULE 4:**9 Hrs****Transport Layer – I** (*Process-to-Process Delivery: UDP, TCP, and SCTP*)

Process-To-Process Delivery: Client/Server Paradigm, Multiplexing and Demultiplexing, Connectionless Versus Connection-Oriented Service, Reliable Versus Unreliable, Three Protocols. User Datagram Protocol (UDP): Well-Known Ports for UDP, User Datagram, Checksum, UDP Operation, Use of UDP. TCP: TCP Services, TCP Features, Segment, A TCP Connection, Flow Control, error control, SCTP: SCTP services, SCTP features.

Transport Layer - II (*Congestion Control and Quality of Service*)

Data traffic , traffic profiles, Congestion: Network Performance. Congestion Control: Open-Loop Congestion Control, Closed-Loop Congestion Control. . Congestion Control in TCP, . Quality of Service: Flow Characteristics, Flow Classes. Techniques To Improve QoS: Scheduling, Traffic Shaping. **Self Learning Exercise:** SCTP packet format.

MODULE 5:**7 Hrs****Application Layer**

Domain Name System: Name Space: Flat Name Space, Hierarchical Name Space. Domain Name Space: Label, Domain Name, Domain. Distribution Of Name Space: Hierarchy of Name Servers, Zone, Root Server, Primary and Secondary Servers. DNS in the Internet: Generic Domains, Country Domains, Inverse Domain. Resolution

Remote Logging, Electronic Mail, and File Transfer: Remote Logging, Telnet, Electronic Mail: Architecture, User Agent, Message Transfer Agent. FTP

Self Learning Exercise: POP3, IMap

TEXT BOOK:

1. Data Communications and Networking, Behrouz A Forouzan, 4th Edition, McGraw-Hill, 2006.

REFERENCE BOOKS:

1. Data and Computer Communication, William Stallings, 8th Edition.
2. Computer Networks: A Systems Approach, Larry L Peterson and Bruce S David, Elsevier, 4th Edition.

EBOOK:

1. <http://www.faadooengineers.com/threads/3176-Computer-Networks-FREE-Ebook-covering-full-semester-syllabus>

MOOCs:

1. <http://www.omnisecu.com/tcpip/tcpip-model.php>
2. <http://www.omnisecu.com/tcpip/index.php>

CO – PO MAPPING:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	M	S			W							
CO 2	M	S			W							
CO 3		S										
CO 4		M										
CO 5		M										

CO – PSO MAPPING:

	PSO1	PSO2	PSO3
CO 1	M	M	
CO 2	M	M	
CO 3	S		
CO 4	S		
CO 5	M		

CRYPTOGRAPHY AND NETWORK SECURITY (3:0:2)**Sub code : IS6C04****Hrs/week : 05****SEE Hrs : 03****CIE : 50 Marks****SEE : 50% Marks****Max. Marks: 100****Course Outcomes:****On Successful completion of the course, the students will be able to:**

1. Explain the concepts of traditional and modern symmetric key ciphers
2. Illustrate modern block ciphers.
3. Describe secure communication using public key algorithms, digital signatures & cryptographic hash functions.
4. Discuss the different key management and authentication protocols.
5. Explain the security aspects at various layers in the network.
6. Demonstrate various cryptographic algorithms (Practical).

MODULE 1: [See Diagrams](#)**8 Hrs****Introduction:** Security Goals, Cryptographic Attacks, **Services and mechanism.****Traditional Symmetric-Key Ciphers:** Symmetric key Ciphers, categories of traditional ciphers, Stream and Block Ciphers.**Introduction to Modern Symmetric-Key Ciphers:** Modern Block Ciphers and its components.**Self Learning Exercise:** Techniques for security goals implementation and Modern Stream ciphers.**MODULE 2:****8 Hrs****Modern Block Ciphers Continued:** Product ciphers and its classes, Attacks designed for block ciphers.**Data Encryption Standard (DES):** DES Structure, Security of DES,**Advanced Encryption Standard (AES):** Transformations, Key expansion, The AES Ciphers. Analysis of AES**Self Learning Exercise:** Multiple DES**MODULE 3:****8 Hrs****Asymmetric-Key Cryptography:** Introduction, RSA Cryptosystems.**Integrity and Authentication:** Message integrity, Message authentication, Digital Signature- Process, Services, Attacks on Digital Signature, DSS.**Cryptographic hash functions:** MD4, SHA-512.**Self Learning Exercise:** Other Digital Signature Schemes, Whirlpool

MODULE 4:**7 Hrs**

Entity Authentication: Entity Authentication and Message Authentication, Password- based Authentication, Challenge-Response Protocols.

Key Management: Symmetric-Key Distribution, Kerberos, Symmetric Key Agreement, Public-Key Distribution.

Self Learning Exercise: Biometrics and Hijacking.

MODULE 5:**8 Hrs**

Security at the Application Layer: PGP, S/MIME.

Security at the Transport Layer: Security service at Transport Layer, SSL Architecture- Four protocols.

Security at Network Layer: IP Security, Modes of IPSec, Two Security Protocols.

System Security: IDS, Firewalls.

Self Learning Exercise: IKE, Malicious programs.

List of Programs:

1. Encrypt and decrypt a file with composite data using the following Traditional symmetric key Ciphers:
 - a) Caesar Cipher
 - b) Playfair Cipher
 - c) Hill Cipher
 - d) Vigenere Cipher
 - e) Rail fence – row & Column Transformation
2. For a given input with alpha-numeric data, encrypt and decrypt using Data Encryption Standard (DES) algorithm
3. Consider an alpha-numeric data, encrypt and decrypt it using Advanced Encryption Standard (AES) algorithm.
4. Encrypt and decrypt a file with composite data using RSA algorithm and demonstrate the cycling attack on RSA.
5. Implement the following cryptographic hash functions
 - a) MD5
 - b) SHA-512
6. Implement the Signature Scheme - Digital Signature Standard to sign and verify a given message.
7. Implement the Diffie-Hellman key exchange algorithm to setup a security channel between client and server.
8. Demonstrate secure data storage and transmission by creating digital signatures using GNUPG (The GNU Privacy Guard)
9. Demonstrate Intrusion Detection System (IDS) using any tool (like SNORT)
10. Demonstrate Packet Filtering Firewalls using any simulation tool (like CISCO Packet Tracer)

TEXT BOOK:

1. Cryptography and Network Security, by Behrouz A Forouzan and Debdeep Mukhopadhyay, 3rd Edition, McGraw-Hill, 2016

REFERENCE BOOKS:

1. Cryptography and Network Security: Principles and Practice, by William Stallings, 7th Edition, Prentice Hall, 2017.
2. Cryptography and Network Security, 3rd Edition, Atul Kahate, 2017.

EBOOKS:

1. <http://www.nptelvideos.in/2012/11/cryptography-and-network-security.html>
2. https://www.cs.ucy.ac.cy/courses/EPL475/Cryptography_and_Network_Security_Principles_and_Practice_5thEdition.pdf

MOOC's:

1. <http://nptel.ac.in/courses/106105031/>
2. <https://www.coursera.org/learn/crypto>

CO – PO MAPPING:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	M	M	S	W		M						
CO 2	M	M	S	S		M	W					
CO 3	M	M	S	S		S						
CO 4	M	M	S	S		S						
CO 5	M	M	S	S	W	S						
CO 6	S	S	S	S	M	M						

CO – PSO MAPPING:

	PSO1	PSO2	PSO3
CO 1	M		
CO 2	S	W	
CO 3	M		
CO 4	M		
CO 5	M		
CO 6	M	M	

ENGINEERING ECONOMICS (3:0:0)**Sub code : IS6C05****Hrs/week : 03****SEE Hrs : 03****CIE : 50 Marks****SEE : 50% Marks****Max. Marks: 100****Course Outcomes:****On Successful completion of the course, the students will be able to:**

1. Recall the basic concepts of decision making, problem solving, tactics and strategy.
2. Explain Micro and Macro Economic terms.
3. Define the time value of money concept, interest formulae.
4. Explain the concepts of depreciation and replacement criteria.
5. Explain estimation process to calculate the total cost of components.

MODULE 1:**8 Hrs****Decision Making** – Introduction: engineering decision – makers, engineering and economics, problem solving, intuition and analysis, tactics and strategy.**Self Learning Exercise:** Computational economics**MODULE 2:****7 Hrs****Micro Economics:** Law of Demand, Law of supply, Market equilibrium and Determination of Price.**Macro Economics:** Consumption goods, Capital goods, Final goods, Intermediate goods; stocks and flows; gross investment and depreciation. Circular flow of income (two sector model); Methods of calculating National Income - Value Added or Product method, Expenditure method, Income method. Aggregates related to National Income: Gross National Product (GNP), Net National Product (NNP), Gross and Net Domestic Product (GDP and NDP)**Self Learning Exercise:** Real and Nominal GDP**MODULE 3:****7 Hrs****Interest and Interest Factors:** Interest rate, simple interest compound interest, interest formulae, time value equivalence exercises, problems and discussion.**Self Learning Exercise:** Cash flow diagrams**MODULE 4:****7 Hrs****Depreciation:** Introduction, methods of depreciation, problems. Replacement Analysis: Reasons- Deterioration, obsolescence, inadequacy, replacement criteria problems.**Self Learning Exercise:** Causes of depreciation, reasons for replacement

MODULE 5:**10 Hrs**

Estimating and Costing: Components of costs such as direct material cost, direct labour cost, Fixed, over – heads, factory costs, administrative – over heads, first cost, selling price, calculation of the total cost of various components, Mensuration, estimation of simple components. Break Even Analysis

Self Learning Exercise: Marginal Cost, Sunk Cost

TEXT BOOK:

1. Engineering economy – Riggs J.L., McGraw Hill, 2002.
2. Engineering economy – Paul Degarmo, Macmillan Pub, Co., 2001.
3. Engineering Economy – Theusen. G. PHI, 2002.

REFERENCE BOOKS:

1. Engineering economics-R. Panneerselvam (2nd Edition), PHI Learning Pvt. Ltd., 2013
2. Industrial Management, D K Bhattacharyya Vikas Publishing
3. Financial Management- I.M. Pandey (9th Edition) “Financial Management”, Vikas publication, 2011

CO – PO MAPPING:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	M	S	M	W		M	W	S	W	W	M	M
CO 2	M					M	M	M	M		M	
CO 3	W	M				W	M		M		S	
CO 4	M					S	M		M	S	S	M
CO 5	W					M	S		M	M	S	S

CO – PSO MAPPING:

	PSO1	PSO2	PSO3
CO 1	M	M	M
CO 2	M	M	M
CO 3	W	M	S
CO 4		S	S
CO 5		W	S

WEB TECHNOLOGIES LAB (0:0:3)

Sub code : IS6L01**Hrs/week : 03****SET Hrs : 03****CIE : 25 Marks****SET : 25 Marks****Max. Marks: 50****Course Outcomes:****On Successful completion of the course, the students will be able to:**

1. Develop client side programming
2. Develop server side programming

List of Programs:

1. Develop and demonstrate a XHTML document that illustrates the use external style sheet, ordered list, table, borders, padding, color, and the tag.
2. Develop and demonstrate a XHTML file that includes Javascript for the following problems:
 - a. Input: A number n obtained using prompt
Output: The first n Fibonacci numbers
 - b. Input: A number n obtained using prompt
Output: A table of numbers from 1 to n and their squares using alert
3. Develop and demonstrate a XHTML file that includes Javascript script that uses functions for the following problems:
 - a. Parameter: A string
Output: The position in the string of the left-most vowel
 - b. Parameter: A number
Output: The number with its digits in the reverse order
4. Develop and demonstrate, using JavaScript, a XHTML document that collects the USN (the valid format is: A digit from 1 to 4 followed by two upper-case characters followed by two digits followed by two upper-case characters followed by three digits; no embedded spaces allowed) of the user. Event handler must be included for the form element that collects this information to validate the input. Messages in the alert windows must be produced when errors are detected.
5.
 - a. Design an XML document to store information about a student in an engineering college. The information must include USN, Name, Name of the College, Branch, Year of Joining, and e-mail id. Make up sample data for 3 students. Create a CSS style sheet and use it to display the document.
 - b. Create an XSLT style sheet for one student element of the above document and use it to create a display of that element.
6. Write a PHP program to display various Server Information like Server Name, Server Software, Server protocol, CGI Revision etc.
7. Write a PHP Program to accept three no.s from command prompt and find the largest number using functions.

8. Write a PHP program to accept the User Name and display a greeting message randomly chosen from a list of 4 greeting messages.
9. Write a PHP program to keep track of the number of visitors visiting the web page and to display this count of visitors, with proper headings.
10. Write a PHP program to store current date-time in a COOKIE and display the 'Last visited on' date-time on the web page upon reopening of the same page.
11. Write a PHP program to store page views count in SESSION, to increment the count on each refresh, and to show the count on web page.
12. Using PHP and MySQL, develop a program to accept book information viz. Accession number, title, authors, edition and publisher from a web page and store the information in a database and to search for a book with the title specified by the user and to display the search results with proper headings.
13. A sample program for creating basic angular application.

Open Ended Experiments:

Students should develop a website about their college / library or any other problem which includes all the features studied in the course. (XHTML, CSS, Javascript, XML, PHP, Mysql and Angular)

TEXT BOOK:

1. Programming the World Wide Web, Robert W Sebesta, 8th Edition, Addison Wesley, 2014

REFERENCE BOOKS:

1. Internet & World Wide Web How to Program, M. Deitel, P.J. Deitel, A. B. Goldberg, 4th Edition.
2. Web Programming Building Internet Applications, Chris Bates, 3rd Edition.
3. The web Warrior Guide to Web Programming, Cengage Learning, Xue Bai.

CO – PO MAPPING:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	M	S	S	M	M							
CO 2	S	S	S	M	M							

CO – PSO MAPPING:

	PSO1	PSO2	PSO3
CO 1	M	S	
CO 2	S	S	

COMPUTER NETWORKS LAB (0:0:3)

Sub code : IS6L02**Hrs/week : 03****SET Hrs : 03****CIE : 25 Marks****SET : 25 Marks****Max. Marks: 50****Course Outcomes:****On Successful completion of the course, the students will be able to:**

1. Implement the concepts of communication in a computer network
2. Demonstrate the computer network concepts using simulators

PART – A

1. Prepare a detailed report on different devices available in various layers of computer network architecture.
2. Given a valid IPV4 address, separate the Network and Host ID part and determine the class of the given IPV4 address. (Using a C program)
3. Simulate the working of sliding window protocol for flow control in computer networks. (Using a C program)
4. Simulate the working of distance vector routing algorithm using Bellman-Ford's algorithm.
5. Write a client-server program in TCP / IP in which the server-side code listens for connect requests, and whatever message the client sends the server converts it to uppercase and sends it back.
6. Write a program to generate a Hamming Code for a four-bit data. The program should also detect and correct error for the received data.

PART – B

1. Using **Wireshark** capture, filter and inspect packets.
 - a. Capture Ethernet / Wi-Fi Packets.
 - b. Filter TCP, UDP, HTTP, DNS Packets.
 - c. Capture filter for telnet that captures traffic to and from a given host.
2. Using **ESTINET** simulator carry out the following exercises (topology will be given during the experiment conduction.)
 - a. Draw a wired network topology (using hub, switch and router) and set the parameters like "Bandwidth", "Bit Error Rate", and "Propagation Delay" and run the simulation to calculate the number of packets dropped.
 - b. Draw a topology of a wireless network involving host, router, obstacle, access point, ad-hoc, infra and multi interface mobile nodes of 802.11 (a/g) and set the parameters of different nodes used. Calculate the number of collisions occurring at a specified node.

OPEN ENDED EXPERIMENTS

1. Simulate and Study of network IP using
 - Classification of IP addresses
 - Subnetting
 - Supernetting
2. Simulate the Connection of computers in Local Area Network for the ring/star topology
3. Connect two virtual machines and demonstrate the communication between the two using TCP protocol

TEXT BOOK:

1. Data Communications and Networking, Behrouz A Forouzan, 4th Edition, McGraw-Hill, 2006.

REFERENCE BOOKS:

1. Data and Computer Communication, William Stallings, 8th Edition.
2. Computer Networks: A Systems Approach, Larry L Peterson and Bruce S David, Elsevier, 4th Edition.

EBOOK:

1. ESTINET GUI manual

CO – PO MAPPING:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	M	S	S	M	W							
CO 2	S	S	S	M	M							

CO – PSO MAPPING:

	PSO1	PSO2	PSO3
CO 1	S	S	
CO 2	M	S	

MINOR PROJECT (0:0:2)

Sub code : IS6C06

Hrs/week : 02

SET Hrs : 03

CIE : 50 Marks

SET : 50 Marks

Max. Marks: 100

Course Outcomes:

On Successful completion of the course, the students will be able to:

1. Identify areas of interest in emerging technology.
2. Formulate the problem and perform analysis.
3. Implement cost effective design methods with documentation

Note: A Project team consists of minimum of three and maximum of four students.

CO – PO MAPPING:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	M	S		W		W	M	W	S		W	
CO 2	M	S	W		M				M		W	
CO 3	M	M	S	M	M				M	M	M	

CO – PSO MAPPING:

	PSO1	PSO2	PSO3
CO 1	S		S
CO 2		W	
CO 3	S	W	M

ELECTIVE - I
DATA SCIENCE AND DATA ANALYTICS (2:0:2)

Sub code : IS6E101

Hrs/week : 04

SEE Hrs : 03

CIE : 50 Marks

SEE : 50% Marks

Max. Marks: 100

Course Outcomes:

On Successful completion of the course, the students will be able to:

1. Explain tools used for data science exploration
2. Identify the factors used for pattern recognition.
3. Discuss regression methods
4. Discuss clustering techniques
5. Use PCA in dimensionality reduction

MODULE 1:

8 Hrs

Introduction

Data Science, Data Scientist, characteristics of a data scientist and data science team, Data science tools, open source tools, Data to insight

Self Learning Exercise: Data science workflow

MODULE 2:

8 Hrs

Machine Learning and Pattern Recognition

Recognizing patterns, AI and machine learning, Data is good but other things to consider, Learning, predicting and classifying, Machine learning and data science, Feature selection, Bias, variance, Regularization, useful measures, distance and similarity, training and testing

Self Learning Exercise: Cross-validation

MODULE 3:

8 Hrs

Regression

Regression, Multivariate linear regression, ordinary least square, brain and body, regression with one variable, Logarithmic transformation, standardization and scaling, polynomial regression, variance-bias tradeoff, shrinkage

Self Learning Exercise: LASSO and ridge

MODULE 4:

8 Hrs

Clustering

Clustering, clustering with k-means, ensemble techniques, bagging, boosting, random forests

Self Learning Exercise: Stacking and blending

MODULE 5:**7 Hrs****Dimensionality reduction**

Dimensionality reduction, Principal component analysis, PCA in the Iris dataset

Self Learning Exercise: Single value decomposition**TEXT BOOK:**

1. Data Science and Analytics by JesusRogel-Salzar, First edition, CRC press, 2017

REFERENCE BOOK:

1. Introducing data science by DAVY CIELEN, ARNO D. B. MEYSMAN, MOHAMED ALI, Manning publication , 2016

MOOCs

1. <https://www.mooc-list.com/categories/statistics-data-analysis>
2. <https://nptel.ac.in/courses/106/106/106106179/>

CO – PO MAPPING:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	S	S			M							
CO 2	M	S	W									
CO 3	M	S	M									
CO 4	M	S		M								
CO 5		M		M								

CO – PSO MAPPING:

	PSO1	PSO2	PSO3
CO 1	S	M	
CO 2	M	S	
CO 3	M	S	
CO 4	M	S	
CO 5	W	M	

CYBER SECURITY (3:0:0)**Sub code : IS6E102****Hrs/week : 03****SEE Hrs : 03****CIE : 50 Marks****SEE : 50% Marks****Max. Marks: 100****Course Outcomes:****On Successful completion of the course, the students will be able to:**

1. Describe cybercrime and its classifications.
2. Explain various types of cyber attacks.
3. Describe security challenges faced by the mobile devices.
4. Distinguish the different tools and methods used in cybercrime.
5. Discuss the impact of Phishing.

MODULE 1:**8 Hrs****Introduction to Objectives:**

Introduction to cybercrime, Cybercrime and information security, who are Cybercriminals, Classification of Cybercrimes, Cybercrime: The Legal Perspectives, An Indian Perspective, Cybercrime and the Indian ITA 2000, A global perspective on cybercrimes.

Self Learning Exercise: Cybercrime era: Survival mantra for the citizens.

MODULE 2:**8 Hrs****Cyber offenses:**

Introduction, How criminal plan the attacks, Social engineering, Cyber stalking, Cybercafe and cybercrimes, Botnets: The fuel for cybercrime, Attack vector

Self Learning Exercise: Cloud Computing.

MODULE 3:**8 Hrs****Cyber crime: Mobile and Wireless devices**

Introduction, Proliferation of mobile and wireless devices, Trends in mobility, Security challenges posed by mobile devices, Registry setting for mobile devices, Authentication service security, Attacks on mobile/ cell phones, Mobile devices: security implications for organizations, Organization measures for handling mobile, Organizational security policies.

Self Learning Exercise: Measures in mobile computing era, Laptops.

MODULE 4:**8 Hrs****Tools and method used in Cybercrime:**

Introduction, Proxy servers and anonymizers, Phishing, Password cracking, Key loggers and spywares, Virus and worms, Trojan horses and backdoors, Steganography, DoS and DDoS attacks, SQL injection, Buffer overflow

Self Learning Exercise: Attacks on wireless networks.

MODULE 5:**7 Hrs****Phishing and identity theft:**

Introduction, Phishing,

Self Learning Exercise: Identity theft (id theft).

TEXT BOOK:

1. Cyber Security by Nina Godbole, Sunit Belapure, Wiley India, 1st edition copyright 2011, reprint 2013.

REFERENCE BOOK:

1. Computer Forensics and Cyber Crime An Introduction by Marjie T. Britz, Pearson publication, 3rd edition, 2013.

EBOOK:

1. Introduction to computer Networks and cybersecurity by Chwan-Hwa, David Irwin, CRC Press, 2013.

MOOCs:

1. <http://www.open.edu/openlearn/futurelearn/cyber-security>
2. <http://www.cyberdegrees.org/resources/free-online-courses/>

CO – PO MAPPING:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	M	M	S	S		M						
CO 2	S	S	M	M								
CO 3	S	M	W	M	M							
CO 4	M	M	S	M	S	W						
CO 5	S		M	M	M	M						

CO – PSO MAPPING:

	PSO1	PSO2	PSO3
CO 1	S	W	
CO 2	M		
CO 3	M	M	
CO 4	S	W	M
CO 5	M	W	

INTRODUCTION TO MACHINE LEARNING (3:0:0)**Sub code : IS6E103****Hrs/week : 03****SEE Hrs : 03****CIE : 50 Marks****SEE : 50% Marks****Max. Marks: 100****Course Outcomes:****On Successful completion of the course, the students will be able to:**

1. Explain the basics of machine learning and its classifications.
2. Illustrate the working of algorithms on classifications.
3. Discuss the working of Support Vector Machines and Decision tree.
4. Illustrate data pre-processing techniques.
5. Describe Data Compression using Dimensionality Reduction techniques.

MODULE 1:**8 Hrs**

Giving Computers the Ability to Learn from Data Building intelligent machines to transform data into knowledge The three different types of machine learning, Making predictions about the future with supervised learning Classification for predicting class labels Regression for predicting continuous outcomes Solving interactive problems with reinforcement learning Discovering hidden structures with unsupervised learning Finding subgroups with clustering Dimensionality reduction for data compression **An introduction to the basic terminology and notations A roadmap for building machine learning systems** Preprocessing – getting data into shape Training and selecting a predictive model Evaluating models and predicting unseen data instances, **Using Python for machine learning**

Self Learning Exercise: Installing Python packages

MODULE 2:**7 Hrs****Training Machine Learning Algorithms for Classification**

Artificial neurons – a brief glimpse into the early history of machine learning Implementing a perception learning algorithm in Python Training a perception model on the Iris dataset.

Adaptive linear neurons and the convergence of learning

Self Learning Exercise: Minimizing cost functions with gradient descent

MODULE 3:**8 Hrs****A Tour of Machine Learning Classifiers Using Scikit-learn**

Choosing a classification algorithm First steps with scikit-learn Training a perception via scikit-learn **Modeling class probabilities via logistic regression** Logistic regression intuition and conditional probabilities Learning the weights of the logistic cost function Training a logistic regression model with scikit-learn Tackling over fitting via regularization.

Maximum margin classification with support vector machines Maximum margin intuition Dealing with the nonlinearly separable case using slack variables Alternative implementations in scikit-learn **Solving nonlinear problems using a kernel SVM** Using the kernel trick to find separating hyper planes in higher dimensional space

Decision tree learning Maximizing information gain – getting the most bang for the buck Building a decision tree Combining weak to strong learners via random forests

Self Learning Exercise: K-nearest neighbors – a lazy learning algorithm

MODULE 4:**8 Hrs****Building Good Training Sets – Data Preprocessing**

Dealing with missing data Eliminating samples or features with missing values Imputing missing values Understanding the scikit-learn estimator API Handling categorical data Mapping ordinal features Encoding class labels Performing one-hot encoding on nominal features

Partitioning a dataset in training and test sets Bringing features onto the same scale Selecting meaningful features Sparse solutions with L1 regularization

Self Learning Exercise: Sequential feature selection algorithms, Assessing feature importance with random forests

MODULE 5:**8 Hrs****Compressing Data via Dimensionality Reduction**

Unsupervised dimensionality reduction via Principal Component Analysis Total and explained variance Feature transformation Principal component analysis in scikit-learn

Supervised data compression via linear discriminant analysis Computing the scatter matrices Selecting linear discriminants for the new feature subspace Projecting samples onto the new feature space LDA via scikit-learn

Using kernel principal component analysis for nonlinear mappings Kernel functions and the kernel trick Implementing a kernel principal component analysis in Python Example 1 – separating half-moon shapes Example 2 – separating concentric circles Projecting new data points

Self Learning Exercise: Kernel principal component analysis in scikit-learn

TEXT BOOKS:

1. “Python Machine Learning”, Machine learning and deep learning with python, Scikit-learn and TensorFlow2, by Sebastian Raschka 3rd edition-includes TensorFlow2, GANs and Reinforcement Learning, 2019
2. Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, by Aurelien Geron O'Reilly Publications, 2nd edition, 2019

REFERENCE BOOKS:

1. Introduction to Machine Learning (Adaptive Computation and machine learning), Ethem Alpaydm, The MIT Press Cambridge, Massachusetts London, ISBN: 0-262- 01211-1, 2004
2. A first course in machine learning, Simon Rogers, Mark Girolami, Chapman, & Hall/CRC machine learning & pattern recognition, 2011
3. FUNDAMENTALS OF MACHINE LEARNING FOR PREDICTIVE DATA ANALYTICS Algorithms, Worked Examples, and Case Studies, John D. Kelleher, Brian Mac Namee, Aoife D'Arcy, The MIT Press, Cambridge, Massachusetts, London, England

EBOOKS:

1. alex.smola.org/drafts/thebook.pdf
2. https://www.mathworks.com/content/dam/mathworks/tag-team/Objects/i/88174_92991v00_machine_learning_section1_ebook.pdf

MOOCs:

1. https://onlinecourses.nptel.ac.in/noc18_cs26/preview
2. [https://www.coursera.org/learn/machine-learning\[stanforduniversity\]](https://www.coursera.org/learn/machine-learning[stanforduniversity])
3. <https://www.edx.org/course/principles-of-machine-learning>

CO – PO MAPPING:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	S	M		W								
CO 2	S	M	M		W							
CO 3	S	M	S	M								
CO 4	S	S	S	M								
CO 5	M	S	S	S	M							

CO – PSO MAPPING:

	PSO1	PSO2	PSO3
CO 1	M	M	
CO 2	M	M	
CO 3	M	S	
CO 4	M	S	
CO 5	M	S	

OBJECT ORIENTED ANALYSIS & DESIGN (3:0:0)

Sub code : IS6E104**Hrs/week : 03****SEE Hrs : 03****CIE : 50 Marks****SEE : 50% Marks****Max. Marks: 100****Course Outcomes:****On Successful completion of the course, the students will be able to:**

1. Identify the functionalities of the Object Oriented models
2. Describe process overview and domain analysis.
3. Analyze common architectural styles.
4. Explain class design and modeling.
5. Analyze various design pattern categories.

MODULE 1:**8 Hrs****Introduction, Modeling Concepts, Class Modeling:**

What is Object Orientation? What is OO development? OO themes, Evidence for usefulness of OO development, OO modeling history. Modeling as Design Technique: Modeling, abstraction, the three models. **Class Modeling:** Object and class concepts; Link and associations concepts; Generalization and inheritance; A sample class model.

Self Learning Exercise: Navigation of class models.

MODULE 2:**8 Hrs**

Process Overview, System Conception, Domain Analysis: Process Overview: Development stages; Development life cycle. System Conception: Devising a system concept; elaborating a concept; preparing a problem statement. Domain Analysis: Overview of analysis; Domain class model; Domain state model; Domain interaction model;

Self Learning Exercise: Iterating the analysis.

MODULE 3:**8 Hrs****Application Analysis, System Design:**

Application Analysis: Application interaction model; Application class model; Application state model; adding operations. Overview of system design; Estimating performance; Making a reuse plan; Breaking a system in to sub-systems; Identifying concurrency; Allocation of sub-systems; Management of data storage; Handling global resources; Choosing a software control strategy; Handling boundary conditions; Setting the trade-off priorities; Common architectural styles.

Self Learning Exercise: Architecture of the ATM system as the example.

MODULE 4:**8 Hrs**

Class Design, Implementation Modeling, Legacy Systems: Class Design: Overview of class design; Bridging the gap; Realizing use cases; Designing algorithms; Recursing downwards, Refactoring; Design optimization; Reification of behavior; Adjustment of inheritance; Organizing a class design; ATM example. Implementation Modeling: Overview of implementation; Fine-tuning classes; Fine-tuning generalizations; Realizing associations; Testing Legacy Systems: Reverse engineering; Building the class models;

Self Learning Exercise: Building the interaction model.

MODULE 5:**7Hrs****Design Patterns:**

What is a pattern and what makes a pattern? Pattern categories; Relationships between patterns; Pattern description. Communication Patterns: Forwarder-Receiver; Client-Dispatcher-Server; Publisher-Subscriber. Management Patterns: Command processor; View handler. Idioms: Introduction; what can idioms provide? Idioms and style; Where to find idioms.

Self Learning Exercise: Counted Pointer example.

TEXT BOOKS:

1. Object-Oriented Modeling and Design with UML Michael Blaha, James Rumbaugh, 2nd Edition, Pearson Education, 2007
2. Pattern-Oriented Software Architecture: A System of Patterns - Volume 1 Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, Michael Stal, John Wiley and Sons, 2006

REFERENCE BOOKS:

1. Object Oriented Analysis and Design with the Unified Process Satzinger, Jackson, Burd, 2005
2. Object-Oriented Analysis and Design with Applications Grady Booch et al 3rd Edition, Pearson Education, 2007.

EBOOKS:

1. www.tutorialspoint.com/object_oriented_analysis_design
2. freebooksonline.net/pdf/object-oriented.

MOOCs:

1. <http://nptel.ac.in/courses/106105153/>

CO – PO MAPPING:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	S	M	W	W								
CO 2	S	M										
CO 3	S	S	M									
CO 4	M	M	S									
CO 5	M	S	S	M								

CO – PSO MAPPING:

	PSO1	PSO2	PSO3
CO 1	S	M	
CO 2	M	M	
CO 3	S	M	W
CO 4	M	S	
CO 5	M	M	

UX & UI (3:0:0)**Sub code : IS6E105****Hrs/week : 03****SEE Hrs : 03****CIE : 50 Marks****SEE : 50% Marks****Max. Marks: 100****Course Outcomes:****On Successful completion of the course, the students will be able to:**

1. Describe the concepts of User interface patterns
2. Illustrate different features, approach and pattern of User Interface / User Experience (Module 2 & 3)
3. Discuss the effects of Wearable devices on health and environment.
4. Describe continuous and complementary design approach in developing interfaces

MODULE 1:**8 Hrs**

What Users Do: A Means to an End, the Basics of User Research, Users' Motivation to Learn, The Patterns – Safe Exploration, Instant Gratification, Satisfying, Changes in Midstream, Deferred Choices, Incremental Construction, Habituation, Micro breaks, Spatial Memory, Prospective Memory, Streamlined Repetition, Keyboard Only.

Self Learning Exercise: Other People's Advice, Personal Recommendations.

MODULE 2:**8 Hrs**

Information Architecture and Application Structure: The Big Picture, The Patterns – Feature, Search and browse, News Stream, Picture Manager, Dashboard, Canvas Plus Palette, Wizard Setting Editor, Alternative Views, Many Workspaces, Multi-Level Help Making it Look Good: Visual Style and Aesthetics: Same content, Different styles, The Basics of Visual Design, What This Means for Desktop Applications, The Patterns: Deep Background, Few Hues, Many Values, Corner Treatments, Borders That Echo Fonts, Hairlines.

Self Learning Exercise: Contrasting Font Weights, Skins and Themes

MODULE 3:**8 Hrs**

Design and UX: Users Vs Life Cycles, Visual Design, Web standards, Potential Barriers to sustainable UX, Designing for Emerging Technologies: Design for Disruption, Eight Design Tenets for Emerging Technology, Changing Design and Designing Change, Fashion with Function: Designing for wearable devices, The next big wave in Technology, The wearable market segments.

Self Learning Exercise: UX (and Human) Factors to consider

MODULE 4:**8 Hrs**

An Ecosystem of connected device: The concept of an Ecosystem, The 3Cs Framework: Consistent, Continuous and Complementary, Single Device Design is History, It's an Eco system, The Consistent Design Approach: What is consistent Design, Consistency in Minimal Interface, Progressive Disclosure in Consistent Design.

Self Learning Exercise: Beyond Device Accessibility

MODULE 5:**7 Hrs**

The Continuous and Complementary Design Approach: The continuous Design Approach: What is Continuous Design? Single Activity flow and the Sequenced Activity Flow. What is Complementary Design? Collaboration: Must-Have, Collaboration: Nice to have, Control: Nice to Have, Fascinating Use Cases: What do they mean for my work? Integrated Design Approaches: 3 Cs as building blocks.

Self Learning Exercise: Beyond the Core Devices: The Internet of Things.

TEXT BOOKS:

1. Jenifer Tidwell, "Designing Interfaces", 2nd Edition, Oreilly, 2015.
2. Jonathan Follet, "Designing for Emerging Technologies- UX for Genomics, Robotics and The Internet of Things", 1st Edition, Oreilly, 2014.
3. Michal Levin, "Designing Multi-Device Experiences", 1st Edition, Oreilly, 2014.
4. Tim Frick, "Designing for Sustainability", 1st Edition, Oreilly 2016.

REFERENCE BOOKS:

1. Ben Shneiderman, Plaisant, Cohen, "Jacobs: Designing the User Interface", 5th Edition, Pearson Education, 2010
2. Unger and Chandler, "A Project Guide to UX Design", 2nd Edition, New Riders, 2012

CO – PO MAPPING:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1		S	S									
CO 2		S	S		S	W						
CO 3			M									
CO 4			M									

CO – PSO MAPPING:

	PSO1	PSO2	PSO3
CO 1	M		
CO 2	S	W	
CO 3	M		
CO 4	M		