VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

SCHOOL OF INFORMATION TECHNOLOGY & ENGINEERING

M.Tech (Software Engineering)-Integrated

(M.Tech-SE)

Curriculum

AY 2022-2023

INDEX

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

VISION STATEMENT OF VELLORE INSTITUTE OF TECHNOLOGY

Transforming life through excellence in education and research.

MISSION STATEMENT OF VELLORE INSTITUTE OF TECHNOLOGY

World class Education: Excellence in education, grounded in ethics and critical thinking, for improvement of life.
Cutting edge Research: An innovation ecosystem to extend knowledge and solve critical problems.
Impactful People: Happy, accountable, caring and effective workforce and students.
Rewarding Co-creations: Active collaboration with national & international industries & universities for productivity and economic development.
Service to Society: Service to the region and world through knowledge and compassion.

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

VISION STATEMENT OF THE SCHOOL OF INFORMATION TECHNOLOGY

- To be a leading school that provides transformative education through qualitative teaching and learning practices.
- To be a centre of excellence in education and research, producing global leaders for improvement of the society.

MISSION STATEMENT OF THE SCHOOL OF INFORMATION TECHNOLOGY

- To provide sound fundamentals, and advances in Information Technology, Software Engineering, Digital Communications and Computer Applications by offering world class curricula.
- To create ethically strong leaders and trend setters for next generation IT.
- To nurture the desire among faculty and students from across the globe to perform outstanding and impactful research for the benefit of humanity and, to achieve meritorious and significant growth.

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

M.Tech (Software Engineering)-Integrated

PROGRAMME EDUCATIONAL OBJECTIVES (PEOs)

1. Graduates will be software practitioners and leaders, who would help solve industry's technological problems.

2. Graduates will be engineering professionals, innovators or entrepreneurs engaged in technology development and research, technology deployment, or engineering system implementation in industry.

3. Graduates will function in their profession with social awareness and responsibility.

4. Graduates will interact with their peers in other disciplines in industry and society and contribute towards research and economic growth of the country.

5. Graduates will be successful in pursuing higher studies in engineering or management.

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

M.Tech (Software Engineering)-Integrated

PROGRAMME OUTCOMES (POs)

POs	Description
1	Having an ability to apply mathematics and science in engineering applications
2	Having a clear understanding of the subject related concepts and of contemporary issues
3	Having an ability to design a component or a product applying all the relevant standards and with realistic constraints
4	Having an ability to design and conduct experiments, as well as to analyze and interpret data
5	Having an ability to use techniques, skills and modern engineering tools necessary for engineering practice
6	Having problem solving ability- solving social issues and engineering problems
7	Having adaptive thinking and adaptability
8	Having a clear understanding of professional and ethical responsibility
9	Having cross cultural competency exhibited by working in teams
10	Having a good working knowledge of communicating in English
11	Having a good cognitive load management [discriminate and filter the available data] skills
12	Having interest in lifelong learning

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

M.Tech (Software Engineering)-Integrated

PROGRAMME SPECIFIC OUTCOMES (PSOs)

PSO1: Apply mathematical concepts to solve computational tasks and relate interdisciplinary solutions through logical reasoning ability.

PSO2: Analyze the frameworks of software engineering and use design methodologies for developing complex software systems using advanced techniques.

PSO3: Acquire the expertise in various core and advanced domains of computing and provide innovative solutions.

Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

M.Tech (Software Engineering)-Integrated

CREDIT STRUCTURE

Category-wise Credit distribution

Sl.No.	Category	Credits
1	University Core (UC)	61
2	Programme Core (PC)	76
3	Programme Elective (PE)	72
4	University Elective (UE)	12
Minimum Total Number of Credits (As per Acad. Council)		221

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

UNIVERSITY CORE							
Course Code	**Course Title**	**L**	**T**	**P**	**J**	**C**	**Remarks**
CHY1701	Engineering Chemistry	3	0	2	0	4	
CHY1002	Environmental Sciences	3	0	0	0	3	Non Credit Course
CSE1001	Problem Solving and Programming	0	0	6	0	3	
CSE1002	Problem Solving and Object Oriented Programming	0	0	6	0	3	
ENG1901/ ENG1902/ ENG1903	Technical English-I Technical English-II Advanced Technical English	0/ 0/ 0	0/ 0/ 0	4/ 4/ 2	0/ 0/ 4	2	
ENG1000 ENG2000	Foundation English1 Foundation English 2	0	0	4	0	2	Non Credit Course
HUM1021	Ethics and Values	2	0	0	0	2	
MAT1011	Calculus for Engineers	3	0	2	0	4	
MAT2001	Statistics for Engineers	3	0	2	0	4	
MGT1022	Lean Start-up Management	1	0	0	4	2	
PHY1701	Engineering Physics	3	0	2	0	4	
PHY1901	Introduction to Innovative Projects	1	0	0	0	1	
SWE1901	Technical Answers for Real World Problems (TARP)	1	0	0	4	2	
SWE1903	Comprehensive Examination	0	0	0	0	1	
SWE1904	Co-op/Capstone Project	0	0	0	0	18	
FLC4097	Foreign Language Course Basket	0	0	0	0	2	
STS5097	Soft Skills	-	-	-	-	8	
EXC4097	Co-Extra Curricular Basket	0	0	0	0	2	Non Credit Course
SWE1902	Industrial Internship	0	0	0	0	1	

PROGRAMME CORE						
Course Code	**Course Title**	**L**	**T**	**P**	**J**	**C**
EEE1019	Foundations of Electrical and Electronics Engineering	3	0	2	0	4
MAT1016	Applied Discrete Mathematical Structures	3	2	0	0	4
MAT2002	Applications of Differential and Difference Equations	3	0	2	0	4
SWE1003	Digital Logic and Microprocessor	3	0	2	0	4
SWE1004	Database Management Systems	3	0	2	0	4
SWE1005	Computer Architecture and Organization	3	0	0	0	3
SWE1006	Theory of Computation	3	0	0	0	3
SWE1007	Programming in Java	3	0	2	4	5
SWE1701	Software Engineering	3	0	0	0	3
SWE2001	Data Structures and Algorithms	3	0	2	0	4
SWE2002	Computer Networks	3	0	2	0	4
SWE2003	Requirements Engineering and Management	2	0	0	4	3
SWE2004	Software Architecture and Design	2	0	0	4	3
SWE2005	Software Testing	3	0	0	4	4
SWE2006	Software Project Management	2	0	0	4	3
SWE2007	Software Construction and Maintenance	2	0	0	4	3
SWE3001	Operating Systems	3	0	2	0	4
SWE3002	Information and System Security	3	0	0	4	4
SWE3004	Software Design and Development Project	0	0	0	0	10

PROGRAMME ELECTIVE						
Course Code	**Course Title**	**L**	**T**	**P**	**J**	**C**
BIT1029	Basic Bioinformatics	3	0	0	0	3
CSE3501	Information Security Analysis and Audit	2	0	2	4	4
CSE3502	Information Security Management	2	0	2	4	4
MAT3001	Advanced Mathematics	3	2	0	0	4
MAT3002	Graph Theory and its Applications	3	2	0	0	4
SWE1002	Optimization Techniques	3	2	0	0	4
SWE1008	Web Technologies	3	0	2	0	4
SWE1009	.Net Programming	3	0	2	0	4
SWE1010	Digital Image Processing	3	0	0	4	4
SWE1011	Soft Computing	3	0	0	4	4
SWE1012	E-Governance	2	0	0	4	3
SWE1013	Multimedia Systems	2	0	0	4	3
SWE1014	Enterprise Resource Planning	2	0	0	4	3
SWE1015	Biometric Systems	2	0	0	4	3
SWE1017	Natural Language Processing	2	0	0	4	3
SWE1018	Human Computer Interaction	2	0	0	4	3
SWE2008	Android Programming	3	0	0	4	4
SWE2009	Data Mining Techniques	3	0	0	4	4
SWE2010	Embedded Systems	2	0	0	4	3
SWE2011	Big Data Analytics	3	0	0	4	4
SWE2012	Software Security	2	0	0	4	3
SWE2013	Advanced Java Programming	3	0	0	4	4
SWE2014	Advanced DBMS	2	0	2	0	3

SWE2015	Mainframe Computing	3	0	0	0	3
SWE2016	Semantic Web Technologies	3	0	0	0	3
SWE2017	Parallel Programming	3	0	2	0	4
SWE2018	Object Oriented Analysis and Design	3	0	2	0	4
SWE2019	Design Patterns	2	0	0	4	3
SWE2020	Software Metrics	2	0	0	4	3
SWE2021	Software Configuration Management	3	0	0	0	3
SWE2022	Software Engineering Process, Tools and Methods	2	0	0	4	3
SWE2023	Automotive Software Engineering	3	0	0	0	3
SWE2024	Software Reuse	3	0	0	0	3
SWE2025	Personal Software Process	3	0	0	0	3
SWE2026	Team Software Process	3	0	0	0	3
SWE2027	Knowledge Management System	2	0	0	4	3
SWE2028	Software Engineering Economics	3	0	0	0	3
SWE2029	Agile Development Process	3	0	0	0	3
SWE2030	Reverse Engineering	3	0	0	0	3
SWE2031	Global Software Engineering	3	0	0	0	3
SWE2032	Knowledge Engineering	3	0	0	0	3
SWE2034	Ruby Programming	3	0	2	0	4
SWE2035	Big Data Technologies	3	0	2	0	4
SWE3003	Sensor Networks	3	0	0	0	3
SWE3005	Software Quality and Reliability	3	0	0	0	3
SWE3006	Advanced Software Testing	3	0	2	0	4
SWE4001	System Programming	3	0	2	0	4
SWE4002	Cloud Computing	2	0	0	4	3
SWE4003	Distributed Computing	3	0	0	0	3
SWE4004	Geographic Information Systems	2	0	0	4	3

SWE4005	Internet of Things	2	0	0	4	3
SWE4006	Real Time Systems	2	0	0	4	3
SWE4007	Storage Technologies	3	0	0	0	3
SWE4008	High Performance Computing	3	0	0	0	3
SWE4009	Linux Programming	3	0	2	0	4
SWE4010	Artificial Intelligence	3	0	0	4	4
SWE4011	Game Programming	3	0	2	0	4
SWE4012	Machine Learning	3	0	2	0	4

EEE1019	Foundations of Electrical and Electronics Engineering	L	T	P	J	C
		3	0	2	0	4
Pre-requisite	**Nil**			**Syllabus version**		
Anti-requisite				v. 1.0		

Course Objectives:

1. To teach the simple problem of DC and AC circuits.
2. To provide the knowledge of digital systems.
3. To study the important concepts of electronics.

Expected Course Outcome:

1. Solve simple DC circuits using mesh and nodal analysis.
2. Describe the RLC components with sinusoidal sources.
3. Perform the various network theorems.
4. Design of combinational circuits and synthesis of logic circuits.
5. Formulate the sequential logic circuits.
6. Utilize the basic concepts of semiconductor devices and circuits.
7. Discuss the overview of communication engineering.
8. Design and Conduct experiments, as well as analyze and interpret data

Module:1	**Fundamental concepts and DC circuits:**	**6 Hours**

Basic circuit elements and sources, series and parallel connection of circuit elements, Ohm's Law, Kirchoff's Laws, Source transformation, Node Voltage Analysis, Mesh Current analysis.

Module:2	**Single phase AC Circuits:**	**6 Hours**

Introduction to AC circuits and concept of phasors for constant frequency sinusoidal sources. Steady state AC analysis of a RL, RC, RLC Series circuits, AC power calculations, Power factor, Series resonance.

Module:3	**Network Theorems (A.C. and D.C) :**	**5 Hours**

Thevenin's and Norton's, Maximum power transfer and Superposition Theorems.

Module:4	**Digital Systems:**	**6 Hours**

Number system, Boolean algebra, Logic circuit concepts, Combinational circuit decoder, Encoder, Multiplexer, Demultiplexer, Half adder, Full adder, Synthesis of logic circuits.

Module:5	**Sequential logic circuits:**	**6 Hours**

Computer organization, Memory types, Flip Flops – SR, D, T, JK, Counters, Shift registers.

Module:6	**Semiconductor devices and circuits:**	**8 Hours**

Conduction in semiconductor materials, principle of operation, V-I characteristics of PN junction diode, Zener diode, BJT, MOSFET, IGBT, half wave rectifier, full wave rectifier, filters, Class A, Class B, Class C Amplifier.

Module:7	**Analog Modulation:**	**6 Hours**

Introduction, Inverting amplifier, Non-Inverting amplifier, Basic application of operational amplifier:

Subtractor, Summing amplifier, Comparator, Integrator, Differentiator, Analog to Digital converter, Digital to Analog converter. **Communication Engineering:** Modulation and Demodulation - Amplitude and frequency modulation.		
Module:8	**Lecture by industry experts.**	**2 Hours**
	Total Lecture hours:	**Hours: 45**
List of Challenging Experiments (Indicative)		
Software Experiments		
1.	Analysis and verification of circuit using Mesh and Nodal analysis	2 hours
2.	Verification of network theorems using Maximum power transfer	2 hours
3.	Analysis of RLC series circuit	2 hours
4	Design of half adder and full adder	2 hours
5.	Single phase half wave and full wave rectifier	2 hours
Hardware Experiments		
1.	Verification of network theorems using Thevenin‘s	2 hours
2.	Regulated power supply using Zener diode	2 hours
3.	Design of a lamp dimmer circuit using Darlington pair	2 hours
4.	Staircase wiring layout for multi-storied building	2 hours
5.	Design and verification of logic circuit by simplifying the Boolean expression	2 hours
	Total Laboratory Hours	**20 hours**
Text Book(s)		
1.	Allan R. Hambley, _Electrical Engineering – Principles & Applications, Pearson Education, First Impression, 6/e, 2013.	
2.	John Bird, _Electrical circuit theory and technology‘, Newnes publications, 4th Edition, 2010.	
Reference Books		
1.	Charles K Alexander, Mathew N O Sadiku, _Fundamentals of Electric Circuits‘, Tata McGraw Hill, 2012.	
2.	David A. Bell, _Electronic Devices and Circuit‘, Oxford press-2008.	
3.	D. Roy Choudhary, Shail B. Jain, _Linear Integrated Circuits‘, 4th/e, New Age International, 2010.	
Mode of Evaluation: CAT / Assignment / Quiz / FAT / Project / Seminar		
Recommended by Board of Studies	**30/11/2015**	
Approved by Academic Council	**39th AC**	**Date** **17/12/2015**

MAT1016	Applied Discrete Mathematical Structures	L	T	P	J	C
		3	1	0	0	4
Pre-requisite	None	Syllabus Version				
		1.0				

Course Objectives:

1. The aim of this course is to motivate the learners for understanding the fundamental concepts in discrete mathematics required for software engineering such as sets, functions, sequences, computing techniques, mathematical logics, proof techniques, graph theoretical approaches, relations, recurrence equations and new structured types.
2. On completion of this course, the students are expected to implement the learned discrete mathematical ideas in realistic projects of software technology, theoretical computer skills, computer algorithms, networks and data structures.

Expected Course Outcome

1. Know the basic properties and operations of sets, sequences and also apply the basic principles of counting, permutations and combinations for realistic problems
2. Recognize the Boolean logic through the truth tables and also prove the results by direct, indirect methods and by mathematical induction
3. Learn the basic concepts of graphs, shortest path algorithms, concepts of trees and minimum spanning tree algorithms
4. Analyse the various relations and also solve the recurrence equations
5. Understand the concepts of structured types, three-valued logic and binary trees. Vector calculus with physical understanding to deal with subjects such as fluid dynamics

Module:1	Sets, Sequences and Counting	7 hours

Operations on Sets and Cardinality – The Pigeonhole Principle – Sequences – The Characteristic Sequence of a Subset – Counting – Number of k-Sequences on an n-Set – Number of k-Permutations on an n-Set – Number of k-Subsets of an n-Set.

Module:2	Boolean Expressions, Logic and Proof	7 hours

Boolean Expressions and Truth Tables – Predicates and Quantifiers – Valid Arguments – Direct and Indirect Proofs – Mathematical Induction.

Module:3	Graphs	7 hours

Basic Terminology of Graphs – Special Graphs – The Concept of Degree – Paths – Circuits – Connectedness – Euler and Hamiltonian Circuits – Matrix Representations of Graphs – Graph Isomorphism – Isomorphic Invariants – Shortest Path Problem.

Module:4	Trees	6 hours

Definition of Trees – Characterizing Trees – Rooted and Binary Trees and Their Properties –

Spanning Tree – Minimum Spanning Trees.		
Module:5	**Relations**	**6 hours**
Relations – Matrix and Digraph of a Relation – Properties of Relations – Order Relations – Matrix and Digraph of a Partial Order – Minimal and Maximal Elements – Relations on Finite and Infinite Sequences.		
Module:6	**Recurrence Equations and Series**	**5 hours**
Recurrence Equations – Solving First Order Linear Recurrence Equations – Solving Second Order Linear Recurrence Equations – Infinite Series – Zeno's Paradoxes.		
Module:7	**Defining New Structured Types**	**5 hours**
Simple Enumerated Types – More Elaborate Types – Self-Referential Types – Parameterized Types – Reasoning About New Types – Three-Valued Logic – Processing Data – Lists – Binary Trees.		
Module:8	**Contemporary Issues**	**2 hours**
Industry Expert Lecture		
	Total Lecture hours:	**45 hours**
Tutorial	• A minimum of 10 problems to be worked out by students in every Tutorial class. • Another 5 problems per Tutorial Class to be given as home work. **Mode:** Individual Exercises, Team Exercises, Online Quizzes, Online, Discussion Forums	**30 hours**

Text Book(s)

1. Mathematics of Discrete Structures for Computer Science, Gordan J. Pace, Springer-Verlag, 2012.
2. Fundamentals of Discrete Math for Computer Science: A Problem-Solving Primer, Tom Jenkyns and Ben Stephenson, Springer-Verlag, 2013.

Reference Books

1. Discrete Mathematics with Applications, Susanna S. Epp, Fourth Edition, BROOKS/COLET, 2010.
2. Discrete Mathematical Structures with Applications to Computer Science, J.P. Trembley and R. Manohar, Tata McGraw Hill, 35th Reprint, 2008.
3. Discrete Mathematics and its Applications, Kenneth H. Rosen, 7th Edition, Tata McGraw Hill, 2012.
4. Discrete Mathematical Structures, Kolman, R.C. Busby and S.C. Ross, 6th Edition, PHI, 2009.
5. Discrete Mathematics, Richard Johnsonbaugh, 8th Edition, Prentice Hall, 2017.

	6. Discrete Mathematics, S. Lipschutz and M. Lipson, McGraw Hill Education (India), 2013. 7. Narasing Deo, Graph theory with application to Engineering and Computer Science, Prentice Hall India 2014.			
Mode of Evaluation				
Digital Assignments, Quiz, Continuous Assessments, Final Assessment Test				
Recommended by Board of Studies	16. 08. 2017			
Approved by Academic Council	No. 47th	Date	05.10.2017	

MAT-2002	Applications of Differential and Difference equations	L	T	P	J	C
		3	0	2	0	4
Pre-requisite	MAT1011 – Calculus for Engineers	Syllabus Version				
		v.1.0				

Course Objectives:

1. To provide a comprehensive coverage at an introductory level to the subject of ordinary differential equations and difference equations to solve engineering application oriented problems.
2. To understand the nuances of Matrix methods, Laplace transform techniques and eigenvalue problems.
3. To introduce Z transform technique to solve Difference equations.

Expected Course Outcome:

1. employ the tools of Fourier series to find harmonics of periodic functions from thetabulated values
2. apply the concepts of eigenvalues, eigen vectors and diagonalisation in linearsystems
3. know the techniques of solving differential equations
4. understand the series solution of differential equations and finding eigen values,eigen functions of Strum-Liouville‘s problem
5. Know the Z-transform and its application in population dynamics and digital signalprocessing
6. demonstrate MATLAB programming for engineering problems

Module:1	Fourier series:	6 hours

Fourier series – Euler‘s formulae – Dirichlet‘s conditions – Change of interval- half range series – RMS value – Parseval‘s identity – Computation of harmonics.

Module:2	Matrices:	6 hours

Eigen values and Eigen vectors – properties of Eigen values and Eigen vectors-Cayley Hamilton theorem –similarity of transformation-orthogonal transformation and nature of quadratic form.

Module:3	Solution of Ordinary differential equations :	6 hours

Linear second order ordinary differential equation with constant coefficients– solutions of homogenous and non-homogenous equations- method of undetermined coefficients –method of variation of parameters- Solutions of Cauchy-Euler and Cauchy Legendre differential equations.

Module:4	Solution of differential equations through Laplace transform and matrix method:	8 hours

Solution of ODEs – Non homogeneous terms involving Heaviside function – Impulse function – Solving non homogeneous system using Laplace transform. Solving non homogeneous first order system of differential equations $(X' = AX + G, X' = AX)$ - Reduction of nth order differential equation to first order system**.**

Module:5	**Strum Liouville Problems and Power Series Solutions:**	**6 hours**
The Strum-Liouville Problem-orthogonality of Eigen functions – Series solutions of differential equation about ordinary and regular singular points-Legendre differential equations – Bessel'sdifferential equations		
Module:6	**Z-Transform:**	**6 hours**
Z-transform-relation between Z-transform and Laplace Transforms – Z-transforms of standard functions – Inverse Z-transforms: by partial fraction method, by convolution method		
Module:7	**Difference Equation:**	**5 hours**
Difference equation-first and second order difference equations with constant coefficients-Fibonacci sequence-solution of difference equations-complementary functions – particular integrals by the method of undetermined coefficients – solution of simple difference equations using Z-transforms.		
Module:8	**Contemporary Issues**	**2 hours**
Industry Expert Lecture		
	Total Lecture hours:	45 hours

Text Book(s)	
1.	Advanced Engineering Mathematics by Erwin Kreyszig, 10th Edition, John Wiley India, 2015.
Reference Books	
1.	Higher Engineering Mathematics by B.S.Grewal, 43rd Edition, Khanna Publishers, India,(2015).
2	Advanced Engineering Mathematics by Michael D. Greenberg, 2nd Edition, Pearson Education, Indian edition (2006).
Mode of Evaluation	
Digital Assignments (Solutions by using soft skills), Continuous Assessment Tests, Quiz, Final Assessment Test.	

List of Challenging Experiments (Indicative)		
1.	Solving Homogeneous differential equations arising in engineering problems	2 hours
2.	Solving non-homogeneous differential equations and Cauchy, Legendre equations	2 hours
3.	Applying the technique of Laplace transform to solve differential equations	2 hours
4.	Applications of Second order differential equations to Mass spring system (damped, undamped, Forced oscillations), LCR circuits etc.	2 hours
5.	Visualizing Eigen value and Eigen vectors.	2 hours
6	Solving system of differential equations arising in engineering applications	2 hours

7	Applying the Power series method to solve differential equations arising in engineering applications			2 hours
8	Applying the Frobenius method to solve differential equations arising in engineering applications			2 hours
9	Visulizing Bessel and Legendre polynomials			2 hours
10	Evaluating Fourier series-Harmonic series			2 hours
11	Applying Z-Transforms to functions encountered in engineering			2 hours
12	Solving Difference equations arising in engineering applications			2 hours
			Total Laboratory Hours	24 hours
Mode of Evaluation:				
Weekly Assessment, Final Assessment Test				
Recommended by Board of Studies	16-08-2017			
Approved by Academic Council	No. 47th	Date	05-10-2017	

SWE1003	Digital Logic and Microprocessor	L	T	P	J	C
		3	0	2	0	4
Pre-requisite	EEE1019	Syllabus version				
		v.1.20				

Course Objectives:

1. Explain various number systems, negative number representation
2. To design and analyze combinational logic circuits and sequential logic circuits
3. To introduce the architecture and operation of typical microprocessors
4. To familiarize the students with the Assembly language programming.

Expected Course Outcome:

1. Perform the conversion among different number systems; Familiar with basic logic gates – AND, OR & NOT, XOR, XNOR; Independently or work in team to build simple logic circuits using basic.
2. Design combinational logics using basic gates. And optimize simple logic using Karnaugh maps
3. Explain sequential logic components: SR Latch, D Flip-Flop and their usage and analyze sequential logic circuits.
4. Understand state table using T-FF,JK-FF SR- and FFD-FFs
5. Explain components used in the sequential designs and Analytics: Registers, Adders, Shifters, and Counters
6. Understand design process digital systems
7. Solve basic binary math operations using the microprocessor.
8. Analyze assembly language programs; select appropriate assemble into machine a cross assembler utility of a microprocessor.

Module:1	Introduction	6 hours

Review of number systems – Logic gates: NAND, NOR gate as universal building blocks – Simplification of four-variable Boolean equations using Karnaugh maps

Module:2	Combinational Logic circuits	6 hours

Half adder, Full adder, Half subtractor, Full subtractor – 4-bit parallel adder and subtractor – 3-bit binary decoder – Decimal to BCD encoder – 8-to-1 multiplexer, 1-to-8 Demultiplexer

Module:3	Sequential Logic Circuits	4 hours

Flip-flops: SR flip-flop, Edge-triggered flip-flops (SR,D,JK and T), Master-slave JK flip-flop – Shift registers (SISO,SIPO,PISO,PIPO)

Module:4	Sequential Logic Design	4 hours

Counter: 4-bit binary asynchronous and synchronous counter – Decade counter (asynchronous and synchronous) – Ring counter, Memories (RAM, ROM, EPROM,FLASH)

Module:5	The 8086 Microprocessor	8 hours

Pin diagram – CPU architecture – Flags-Interrupts – Instruction Set – Addressing mode

Module:6	8086 Microprocessor and Interfacing	8 hours

Segmentation- Minimum mode maximum mode operations – Memory Interfacing-I/O interfacing

Module:7	Programming model of 8086	7 hours
Programming model of 8086, Assembler directives and Assembly language Programming of 8086		
Module:8	**Contemporary issues**	**2 hours**
Total Lecture hours: # Mode: Flipped Class Room, [Lecture to bevideotaped], Use of physical and computermodels to lecture, Visit to Industry		45 hours

Text Book(s)

1. Ramesh Gaonkar, Microprocessor Architecture, Programming, and Applications with the 8085, Sixth Edition, Penram International Publishing, 2013.

2. Morris Mano, Digital logic and Computer design, 4th Edition, Pearson, 2008

Reference Books

1. Yu-Cheng Liu, Glenn A. Gibson, Microcomputer Systems: The 8086/8088 Family-Architecture Programming and Design, Second Edition, Pearson, 2015.
2. R.K. Gaur, Digital Electronics and Microcomputers, Dhanpat Rai Publications, 2012.
3. Douglas V. Hall, Microprocessors and Interfacing, Revised Second Edition, Tata McGraw- Hill, 2006

Laboratory exercises

Digital Logic Design

To understand and implement the following

1. Basic Logic Gates
2. Combinational Circuits
3. Adders and Subtractors
4. Code Convertors
5. Parallel Adder and Magnitude Comparator
6. Decoder and Encoder
7. Multiplexer and De-multiplexer
8. Sequential Circuits and Shift registers
9. Counters

Microprocessors

1. To write programs in Assembly Language using 8086 instruction set.
2. To perform interfacing of RAM chip
3. To perform interfacing of keyboard controller
4. To perform interfacing of DMA Controller
5. To perform interfacing of UART/USART

Sample Exercises

1. Assume a large room has 3 doors and a switch near each door controls a light in the

room. The light is turned on or off by changing the state of any one of the switches. More specifically the following should happen:

1. The light is OFF when all 3 switches are open.
2. Closing any one switch will turn the light ON.
3. Then closing the second switch will have to TURN OFF the light.
4. If the light is OFF when the 2 switches are closed, then by closing the third switch the light will TURN ON.

2. Design hardware that implements the following pseudo-code using the provided Comparator, Adder and Registers, along with as many multiplexers and de-multiplexers as needed. The comparator has two inputs In1 and In2, and three outputs, C1, C2, and C3. If In1 < In2, C1 = 1; if In1 = In2, C2=1; if In1 > In2, C3 =1 (for a given In1 and In2, only one of the comparator outputs can be 1). The Adder takes as inputs two numbers p and q, and produces an output Sum. There are 5 registers for storing the 5 variables, A, B, X, Y, and Z. • Hint: You do not need to use truth table or K-maps. Insert the muxes/demuxes as appropriate, and show the signal connections from the input registers A, B, X to the output registers Y and Z, through the muxes, comparator , adder, and demuxes. Be sure to show the equations for the select lines of the multiplexers/demultiplexers in terms of the comparator outputs, C1, C2, and C3.
Pseudo-code:
If A<B then
Z= X+ A
Else if A=B then
Z= X+ B
Else
Y = A + B

3. Design a simplified traffic-light controller that switches traffic lights on a crossing where a north-south (NS) street intersects an east-west (EW) street. The input to the controller is the WALK button pushed by pedestrians who want to cross the street. The outputs are two signals NS and EW that control the traffic lights in the Ns and EW directions. When NS or EW are 0, the red light is on, and when they are 1, the green light is on. When there are no pedestrians, NS=0, EW=1 for a minute, follow by NS=1 and EW=0 for 1 minutes, and so on, when WALK button is pushed, Ns and EW both become 0 for a minute when the present minute expires. After that the NS and EW signals continue alerting. For this traffic-light controller a) Develop a state diagram. (Hint: can be done using 3 states) b) Draw the state transition table. C) Encode the states using minimum number of bits. D) Derive the logic schematic for a sequential circuit which implements the state transition table.

4. Many game shows use a circuit to determine which of the contestants ring in first. Design a circuit to determine which of two contestants rings in first. It has two inputs S1 and S0 which are connected to the contestants' buttons. The circuit has two outputs Z1 and Z0 which are connected to LED's to indicate which contestant rang in first. There is also a reset button that is used by the game show host to asynchronously reset the flip-flops to the initial state before each question. If contestant 0 rings in first, the circuit turns on LED 0. Once LED 0 is on, the circuit leaves it on regardless of the inputs until the circuit is asynchronously reset by the game show host. If contestant 1 rings in first, the circuit turns on LED 1 and leaves it on until the circuit is reset. If there is a tie, both LED's are turned on. The circuit requires four states: reset,

contestant 0 wins, contestant 1 wins, and tie. One way to map the states is to use state 00 for reset, state 01 for contestant 0 wins, state 10 for contestant 1 wins, and state 11 for a tie. With this mapping, the outputs are equal to the current state, which simplifies the output equations.

5. Design a simple circuit that could operate a car alarm. The circuit has one input Y which would be connected to the car's door switch to determine if the car door is open or shut. When the door is shut Y = 0, and when the door is open Y = 1. The circuit has one output Z which is used to operate a horn by shorting the wires that go to the horn switch in the steering wheel. When Z = 1, the switch is activated and the horn honks. The circuit would be asynchronously reset by the accessories power line that is high when the ignition is turned on or is in accessory-only mode, both of which require the key to the car.

6. Design a 12 hour Digital clock which is usually set up to start at 12:00, and they count 12:01, 12:02, 12:03, 12:04, 12:05, 12:06, 12:07, 12:08, 12:09, 12:10, and eventually the clock gets to 12:58, 12:59, 1:00, and so on. The one's place of the minutes (the right-most digit) counts 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, and then repeats. The ten's place of the minutes (second digit from the right) counts 0, 1, 2, 3, 4, 5, and then repeats. The hour counter counts 12, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, and repeats.

Microprocessor Based Design Experiments

7. Design a Microprocessor based combinational lock which has a combination of five digits. The five digits are entered from a keyboard and they are to be entered within a 10 seconds. If the right combination is entered the lock will open. If after 10 seconds either all five digits are not entered or a wrong combination is entered then the display will show an error message. Then the system will allow 5 seconds for the first digit to be entered the second time. If after this time the digit is not entered, the system will turn ON the alarm. If the second try fails, the alarm is also turned ON. Then to reset the system the power has to be turned OFF.(Scrambling Keypad)

8. Design a microprocessor based Smart Pill Box Alarm System for Elderly people. The system will alert the user 3 times per day for taking up the pills. The user has to set the system into fixed slots: for example: Morning, Afternoon, Evening and Night. The system will deliver a display message such as –Take this Pill X –five minutes before the scheduled time. A real time clock is to be included in the system to display the current time and will show the alarm as per the time slots.

9. Design an intelligent system for the following real time situation.

   Consider you are driving a car. You are having a limited display area, where you need to display the fuel status, temperature status, Speed limit, Gear Position based on the priority which suits the following context.—There is an obstacle at a distance of 100m and the same is sensed by a sensor. Based on the sensor input, the display has to be displayed to indicate the function to be performed by the driver.‖

10. An event sequence recorder has to be designed for a hospital in your city which will monitor a patient's pulse rate, blood pressure, body temperature. The equipment accepts inputs from different sensors, and prints the sequence in which they operate. It

scans the inputs every millisecond and prints in a compact, type of event (normal or abnormal) and time of occurrence. It also communicates these events over an RS232C link to a remote computer. A real-time clock is included. Design the processor unit using 8086.

11. Elderly users often forget their daily routines. Hence you need to design a microprocessor based unit to help them remember their monthly expenses and bill payments. For example, their house rent, telephone bills, electricity bills, gas requirement, etc. An alarm has to be blown to remind them and when they reset it, it is understood that they have paid and the expense has to be calculated for the entire month and at the end of the month the total expense has to be intimated.

12. Let say that you work in VIT. Each day there is a rush hour in lunch time – everyone wants to get in the food line first. Your school is at the top floor and only way to get to the lobby is to use a lift. So, you call the lift and wait... and wait. Your waiting time could be infinite because everyone in bottom floors are loading the lift, so it never reaches the top! And when it finally does, your lunch time is over. Design a system to overcome this infinite waiting time.

	Recommended by Board of Studies		4-12-2015	
	Approved by Academic Council	No. 39th	Date	17-12-2015

SWE1004	Database Management Systems	L	T	P	J	C
		3	0	2	0	4
Pre-requisite	None	Syllabus version				
		v. 1.20				

Course Objectives:

1. To study the salient features of database systems and the design process at conceptual and logical level.
2. To implement the database design using relational algebra and SQL.
3. To know the supporting subsystems of DBMS

Expected Course Outcome:

1. Compare the file system and DBMS, and know DBMS architecture and classification.
2. Understand conceptual database design
3. Explain the relational model and Write Queries in relational algebra
4. Create and manipulate the database using SQL and write routines using PL/SQL
5. Evaluate the design of database.
6. Read or write made in the database by single user, multiple user and during failures.
7. Execute a query behind the scene and physical design
8. Design ER model and Implement it using SQL and PL/SQL

Module:1	Fundamental Concepts and Architecture:	4 hours

Introduction to database system, Characteristics of the Database Approach, Actors on the Scene, Workers behind the Scene, Advantages of Using the DBMS Approach, Data Models, Schemas, and Instances, Three-Schema Architecture and Data Independence, Database Languages and Interfaces, The Database System Environment, Classification of Database Management Systems

Module:2	Conceptual Database Design	6 hours

High-Level Conceptual Data Models for Database Design, Entity Types, Entity Sets, Attributes, and Keys, Relationship Types, Relationship Sets, Roles, and Structural Constraints, Weak Entity Types, ER Diagrams, Naming Conventions, and Design Issues, Relationship Types of Degree Higher than Two, Enhanced ER, Specialization, Generalization

Module:3	Relational Database Design	8 hours

Relational Model, Constraints, Update Operations and Dealing with Constraint Violations, Database Design Using ER-to-Relational Mapping and EER to Relation, Relational Algebra, Unary Relational Operations, Operations from Set Theory, Binary Relational Operations, Additional Relational Operations

Module:4	Structured Query Language	8 hours

Data Definition and Data Types, Specifying Constraints in SQL, Basic Retrieval Queries in SQL, INSERT, DELETE, and UPDATE Statements in SQL, Virtual tables Inbuilt functions, Complex Queries-nested, correlated, PL/SQL block, cursor, function, procedure, trigger

Module:5	Normalization Theory	5 hours

Informal Design Guidelines for Relation Schemas, Functional Dependencies, Inference Rules, Equivalence, and Minimal Cover, Properties of Relational Decompositions, Algorithms for Relational Database Schema Design, Normal Forms Based on Primary Keys, Boyce-Codd Normal Form

Module:6	**Transaction, Concurrency, Recovery**	**6 hours**
Introduction to Transaction Processing, Desirable Properties of Transactions, Characterizing Schedules Based on Recoverability, Characterizing Schedules Based on Serializability, Concurrency, Two-Phase Locking Techniques for Concurrency Control, Concurrency Control Based on Timestamp Ordering, Multiversion Concurrency Control Techniques, Recovery Concepts, NO-UNDO/REDO Recovery Based on Deferred Update, Recovery Techniques Based on Immediate Update, Shadow Paging, ARIES Recovery Algorithm		
Module:7	**Query Processing and Indexing:**	**6 hours**
Query Execution plan, Basic algorithms for query execution, Heuristic Query Optimization technique, sparse and dense index, primary, secondary and clustered index, B Tree Vs Hash Index.		
Module:8	**Contemporary issues**	**2 hours**
Total Lecture hours: # Mode: Flipped Class Room, [Lecture to be videotaped], Slides, Demonstration of using Oracle- SQL, 2 hrs lectures by industry experts, Evaluation based on Continuous Assessment Test (30%) and Assignments(20%)		**45 hours**

Text Book(s)

1.	Fundamentals of Database Systems by Ramez Elmasri and Shamkant B.Navathe Pearson Education,2013

Reference Books

1. Database Management Systems by Raghu Rama Krishnan, Tata Mcgraw Hill,2010
2. Database System Concepts by Abraham Silberschatz, Henry F.Korth and S.Sudarshan, Tata Mc Graw Hill, 2011
3. Database System Design and Implementation by Rob Cornell,cennage learning, 2011

List of Challenging Experiments (Indicative)

1.	SQL –Creating tables	
2.	SQL- Inserting, deleting, updating tables, Alter table	
3.	SQL –Querying table-simple queries	
4.	SQL- Creating constraints	
5.	SQL- Altering constraints	
6	SQL- In built functions	
7	SQL – Select statements(with different clauses)	
8	SQL- Querying table-complex(nested, correlated)	
9	SQL – Top N Queries ,catalog Queries, views	

10	PLSQL- block, cursor			
11	PLSSQL- trigger			
12	PLSQL- Function, Procedure			
13	SQL-Creating and Querying-type, varray, nested table			
14	API- Creating API for retrieving data from database			
15	API- Creating API for executing procedure/function			
Total Laboratory Hours				30 hours
Recommended by Board of Studies	5-3-2016			
Approved by Academic Council	No. 40th	Date	18-3-2016	

SWE1005	Computer Architecture and Organization	L	T	P	J	C
		3	0	0	0	3
Pre-requisite	EEE1019	Syllabus Version				
		v.1.0				

Course Objectives:

1. To Introduce organizational and architectural aspects of a Digital Computer
2. To explain the function of each element of a memory hierarchy.
3. To familiarize with latest technologies of memory, I/O, ALU design

Expected Course Outcome:

1. Basic organization of computer assembly language program for given task andcontrol unit operations and instruction level parallelism
2. Demonstrate and perform computer arithmetic operations on integer and real numbers.
3. Demonstrate and perform computer arithmetic operations on higher order functions.
4. Categorize memory organization and explain the function of each element of amemory hierarchy.
5. Identify, use and evaluate the storage management policies with respect to differentstorage management.
6. Compare different methods for computer I/O operations
7. Describe and different the device subsystems.
8. Understand emerging trends in Computer Architecture and Organization

Module:1	FUNDAMENTALS OF COMPUTER ARCHITECTURE	7 hours

Organization of the von Neumann machine; Instruction formats; pipelining-The fetch/execute cycle, instruction decoding and execution; Registers and register files; Instruction types and addressing modes; Subroutine call and return mechanism; Other design issues.

Module:2	COMPUTER ARITHMETIC	6 hours

Data Representation, Hardware and software implementation of arithmetic unit for common arithmetic operations: addition, subtraction, multiplication, division(Fixed point and floating point)-floating point IEEE standards

Module:3	DATA REPRESENTATION	5 hours

Conversion between integer and real numbers- rounding and truncation; The generation of higher order functions from square roots to transcendental functions; Representation of non-numeric data (character codes, graphical data)

Module:4	MEMORY SYSTEM ORGANIZATION AND ARCHITECTURE	6 hours

Memory systems hierarchy; Coding, data compression, and data integrity; Electronic, magnetic and

optical technologies; Main memory organization, Types of Main memories, and its characteristics and performance; Latency, cycle time, bandwidth, and interleaving; Cache memories (address mapping, line size, replacement and write-back policies)		
Module:5	**VIRTUAL MEMORY**	**4 hours**
Virtual memory systems-paging, segmentation, address mapping, page tables, page replacement algorithms; Reliability of memory systems; error detecting and error correcting systems		
Module:6	**INTERFACING AND COMMUNICATION**	**8 hours**
I/O fundamentals: handshaking, buffering; I/O techniques: programmed I/O, interrupt-driven I/O, DMA; Buses: bus protocols, local and geographic arbitration. Interrupt structures: vectored and prioritized, interrupt overhead, interrupts and reentrant code		
Module:7	**DEVICE SUBSYSTEMS**	**7 hours**
External storage systems; Organization and structure of disk drives and optical memory; Flash memories, Basic I/O controllers such as a keyboard and a mouse; RAID architectures; I/O Performance; SMART technology and fault detection		
Module:8	**Contemporary issues**.	**2 hours**
	Total Lecture hours	**45 hours**

Text Book(s)	
1.	J. L. Hennessy & D.A. Patterson, Computer architecture: A quantitative approach, Fifth Edition, Morgan Kaufman, 2011
Reference Books	
1	W. Stallings, Computer organization and architecture, Seventh Edition, Prentice-Hall,2005.
2	M. M. Mano, Computer System Architecture, Third Edition, Prentice-Hall 1992.
3	J. P. Hayes, Computer architecture and Organization, Third edition, McGraw Hill, 2002.

Recommended by Board of Studies		5-3-2016	
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE1006	Theory of Computation	L	T	P	J	C
		3	0	0	0	3
Pre-requisite	**MAT1013/MAT1016**	**Syllabus version**				
		v. 1.0				

Course Objectives:

1. Describe mathematical models of computation along with their relationships with formal languages
2. Discuss regular languages and context free languages which are crucial to understand how compilers and programming languages are built
3. Comprehend that not all problems are solvable by computers and some problems do not admit efficient algorithms
4. Interpret rigorous mathematical reasoning skills

Expected Course Outcome:

1. Demonstrate knowledge of basic mathematical models of computation and their relationalship with to formal languages.
2. Identify different type of Finite Automata and their capabilities.
3. Analyze Regular Language and Context Free Grammar
4. Create push down automata for a given language
5. Discuss the abstract models of Turing machine and its types
6. Create modern techniques to solve P,NP,NP hard and NP complete problems
7. Recognize whether a problem is decidable or undecidable

Module:1	Introduction	6 hours

Alphabets, Strings and Languages and Grammars.
Finite Automata – Deterministic Finite Automata (DFA), Language of a DFA, Non-Deterministic Finite Automata (NFA), Language of a NFA. Equivalence of DFA‘s and NFA‘s, NFA with epsilon-transitions, Removing epsilon-transitions from NFA, DFA state minimizations, Equivalence of two DFA‘s.

Module:2	Regular Language and Regular Grammar	6 hours

Regular Expression, Algebraic laws for Regular Expressions, Converting Regular expression to Finite Automata, Converting FA‘s to Regular Expression, Regular grammar – Right and Left linear Grammar, Finite Automata to Regular grammar, Regular grammar to Finite Automata. Regular language closure properties – union, intersection, concatenation, reversal, complement andstar closure, Non-regular Languages – Proving non-regularity with Pumping lemma. Mealy and
Moore machine

Module:3	Context Free Grammar (CFG)	6 hours

Formal definition, Context Free Language (CFL), Leftmost and Rightmost derivations, Parse tree, Ambiguity in grammars and Languages. Simplification of CFG‘s – Removing useless symbols, epsilon-Productions, and unit productions, Normal forms –CNF and GNF

Module:4	**Pushdown Automata (PDA)**	**6 hours**
Formal Definition, Instantaneous Description of PDA's, PDA and CFL. The language of PDA – Acceptance by Final State, Acceptance by Empty Stack. Deterministic Push down automata (DPDA), DPDA's and Regular Languages, DPDAs and CFL's. Pumping lemma for CFL's. Closure properties of CFL's – union, concatenation, Kleene closure, substitution, reversal, intersection with regular set etc..		
Module:5	**Turing Machine**	**6 hours**
Formal definition, Instantaneous Description, Transition diagrams, Language of a Turing Machine, Turing Machine as Language accepters, Turing Machine as Transducer, Variants of TM's – Multi tape TM, Multidimensional TM, Nondeterministic TM. Equivalence of the various variants with the basic model Church-Turing Thesis.		
Module:6	**Recursive and recursively enumerable languages**	**6 hours**
Recursive and recursively enumerable languages, Properties of recursive and recursively enumerable languages, A language that is not recursively enumerable, Unrestricted Grammar, Context- sensitive language, Linear Bounded automata, Chomsky Hierarchy		
Module:7	**Un-decidability**	**7 hours**
Rice's Theorem, Universal Turing Machine, Turing Machine Halting Problem, Post Correspondence Problem. Undecidable problem for CFG, Undecidable problem for Recursive Enumerable Language. Complexity Classes – P,NP,NP Complete, NP Hard and P ≠ NP		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours	**45 hours**

Text Book(s)

1. Hopcroft, John E., Rajeev Motwani, and Jeffrey D. Ullman. Introduction to Automata Theory, Languages and Computation. Boston: Pearson Addison-Wesley, 2013.

Reference Books

1. Peter Linz, An Introduction to Formal Languages and Automata, Jones & Bartlett Publishers, 2011.
2. Sipser, Michael. Introduction to the Theory of Computation. Australia: Course Technology Cengage Learning, 2013.
3. D'Souza, Deepak, and P. Shankar. Modern Applications of Automata Theory. Singapore: World Scientific, 2012.

Recommended by Board of Studies	5-3-2016		
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE1007	Programming in Java	L	T	P	J	C
		3	0	2	4	5
Pre-requisite	CSE1002	Syllabus version				
		v.1.0				

Course Objectives:

1. To understand fundamentals of programming such as variables, conditional and iterative execution, methods, etc.
2. To Understand fundamentals of object-oriented programming in Java including defining classes, invoking methods using class libraries etc.
3. To learn to use java in variety of technology and on different platforms
4. Be able to use the Java SDK environment to create, debug and run simple Java programs.

Expected Course Outcome:

1. Design simple java programs for specific problems
2. Solve problems using object oriented approach and debug the java application using SDK environment
3. Develop application using inheritance and interfaces.
4. Design and develop Graphical user interface using Applets and Swing/ AWT concepts.
5. Incorporate the cutting-edge frameworks for improving the coding designs using JDBC connectivity
6. Build Java application using multithreading and multitasking.
7. Integrate the connectivity among the terminals are implemented using networking concepts
8. The ability to work effectively in a development of any java application using current trend of Java advancements

Module:1	Introduction to OOPS concepts and their implementation in	8 hours

What is Java?-Features of Java-C, C++ vs. Java first program – Constants, variables, data types, operators, expressions, decision making and branching. OOPs concept, its properties and uses-Defining methods and variables in Java-Class Member and instance members- -Role of Constructor- Memory management using Garbage collector-Creating 1D and 2D Arrays-Overloading and Overriding-Use of this and super keywords-Inheritance- Use of final Keywords-Dynamic Binding and Static Binding-Runtime Polymorphism and its power-Abstract class and Interfaces-Inner/Nested classes and its uses.

Module:2	Stream based I/O in Java and String handling	6 hours

Introduction to Input Output-Introduction and Implementation of Byte stream, Character stream, Buffered stream, Data stream, Object stream and File I/O. String class- String Buffer class-String Builder class-String Tokenizer class

Module:3	Packages and Exception Handling	6 hours
Introduction of Package-Programs related to Packages-Scope of Access Modifiers. Introduction of exception and Error-Throwable class-Try, catch and finally blocks-throw and throws keywords-checked and unchecked exceptions-user defined exception		
Module:4	**Multithreading**	**6 hours**
What is Program, Process, Thread?-Multiprocessing, Multithreading and Multitasking-Use of sleep() and suspend methods-Integrated Thread-Synchronization-Use of wait(), notify() and notifyAll() methods		
Module:5	**Database connectivity**	**5 hours**
What is JDBC API?-Driver types-Two-tier and Three-tier models-Connection overview-Statement overview-Sending Batch updates-Result Set overview-Concurrency types-Prepared Statement overview-Callable Statement overview		
Module:6	**Introduction to Applet and Japplet**	**6 hours**
Getting started with Applets-Defining an Applet subclass-Life cycle of an Applet-Applet‘s execution environment-Developing an Applet-Deploying an Applet-Learning Swing with the Netbeans IDE. Introduction to JFC and Swing, Features of the Java Foundation Classes, Swing API Components, Jcomponent Class, Windows, Dialog Boxes, and Panels, Labels, Buttons, Check Boxes, Menus, Toolbars, Implementing Action interface, Pane, JscrollPane, Desktop pane, Scrollbars, Lists and Combo Boxes, Text-Entry Components, Colors and File Choosers, Tables and Trees, Printing with 2D API and Java Print Service API		
Module:7	**Networking**	**6 hours**
Introduction to Networking in Java-What is TCP and UDP-What is Socket and Port-Implementation of Socket and InetAddress class-URL in terms of Java networking programming-Datagram in network environment-Retrieve the IP address from Host Name, vice-versa.		
Module:8	**Contemporary issues:**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)	
1.	JAVA 2: The Complete Reference‖, Herbert Schildt, 9th Edition, TMH, 2014

Reference Books	
1. 2.	Think Java – How To Think Like A Computer Scientist‖ by Allen B. Downey’s 2012 . Thinking In Java‖ Bruce Eckel’s by Prentice Hall, PTR Prentice-Hall Inc 1998.3. Douglas V. Hall.

List of Challenging Experiments (Indicative)		
1.	Basic Programs	
2.	String Handling	
3.	Classes and Objects	
4.	Inheritance	
5.	Exception Handling	
6.	Multithreading	
7.	Packages and Interfaces	
8.	Applets	

9.	JDBC			
10	Networking			
			Total Laboratory Hours	30 hours
Recommended by Board of Studies				5-3-2016
Approved by Academic Council	No. 40th	Date	18-3-2016	

SWE1701	Software Engineering	L	T	P	J	C
		3	0	0	0	3
Pre-requisite	None	Syllabus version				
		v.1.0				

Course Objectives:

1. To introduce the fundamental concepts of Software Engineering
2. To analyse different metrics for efficient software project management.
3. To explain different methods and models for system design

Expected Course Outcome:

1. Understand the best practices and standards and their applications.
2. Analyze a problem, identify and define the user and system requirements.
3. Design a software system and its process to meet user needs
4. Evaluate and select and software systems considering user needs.
5. Evaluate processes and products against the applicable standards and metrics
6. Assist in the creation of an effective project plan.
7. Analyze software risks and identify mitigation strategies.

Module:1	An Overview of Software Engineering:	6 hours

Nature of Software, Software Engineering, Software Process, Software Engineering Practice, Software Process Models: Linear, RAD, Incremental, Spiral Component-based development, Fourth Gen Techniques.

Module:2	Modeling (Requirements)	6 hours

Requirements Engineering, Establishing the Groundwork, Eliciting Requirements, Developing Use Cases, Building the Requirements Model, Negotiating Requirements, Validating Requirements.

Module:3	Modeling (Design)	5 hours

Design within the context of Software Engineering, Design Process, Design Concepts, Design Model-Software Architecture.

Module:4	Software Testing	6 hours

Strategic Approach to Software Testing, Strategic Issues, Test Strategies for Conventional Software, Software Testing Fundamentals, Black box Testing, White box testing.

Module:5	Process and Product Metrics	6 hours

Product Metrics, Metrics for the Requirements Model, Metrics for the Design Model – Architectural Design Metrics, Object-Oriented Design, Software Measurement, Metrics for Software Quality.

Module:6	Managing Software Projects	6 hours

People, Product, Project, Process, Software Project Estimation, Decomposition Technique,

Empirical Estimation Models, Project Scheduling.		
Module:7	**Risk Management and Software Maintenance**	**8 hours**
Software Risks, Risk Identification, Risk Projection, Risk Refinement, Risk Mitigation, Monitoring and Management, RMMM Plan, Software Maintenance, Software Supportability, Re-engineering.		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)	
1.	Roger Pressman, Software Engineering: A Practitioner's Approach, 7th Edition, McGraw-Hill, 2010.
Reference Books	
1.	Ian Sommerville, Software Engineering, 9th Edition, Addison-Wesley, 2010
2.	Pankaj Jalote, A Concise Introduction to Software Engineering, Springer,2008
3.	William E. Lewis , ―Software Testing and Continuous Quality Improvement‖, Third Edition, Auerbach Publications, 2008

Recommended by Board of Studies	12-8-2017		
Approved by Academic Council	No. 47th	Date	5-10-2017

SWE2001	Data Structures and Algorithms	L	T	P	J	C
		3	0	2	0	4
Pre-requisite	CSE1001	Syllabus version				
		v.1.0				

Course Objectives:

1. To understand the basic concepts of data structures and algorithms in various fields.
2. To learn sorting of and search data items.
3. To comprehend the necessity of time complexity in designing algorithms.
4. To design algorithms to solve real life problems

Expected Course Outcome:

1. Analyze and understandings stack operations and its applications in real world problems.
2. Understand the pros and cons of various queues and its operations
3. Demonstrate linear data structures using dynamic arrays
4. Evaluate algorithms and data structures in terms of time and memory complexity of basic operations.
5. Understand, analyze and design sorting and searching algorithms
6. Understand the importance of hashing
7. Design non-linear data structure operations in real world problems
8. Apply suitable data structures and algorithms for autonomous realization of simple programs or program parts

Module:1	Stack	6 hours

Operations on stack, array implementation of stack, applications of stack-balance of parenthesis in algebraic expressions, converting expressions from infix to postfix or prefix form , evaluating postfix or prefix form, Towers of Hanoi problem

Module:2	Queue	6 hours

Operations on queue , circular queue, array implementation of queue, applications of queue

Module:3	List	6 hours

Singly linked list, doubly linked list, circularly singly linked list, operations on linked lists, Linked representation of stack, Linked representation of Queue

Module:4	Algorithm Analysis	6 hours

Asymptotic notations, Abstract data type, growth rate of functions, running time complexity, best, average and worst case analysis – examples

Module:5	Sorting and Searching	6 hours

Bubble sort, insertion sort, selection sort, radix sort, merge sort, quick sort, heap sort, Shell sort, linear search, binary search, time complexity analysis of sorting and searching algorithms.

Module:6	Hashing	6 hours

Hash functions, open hashing-separate chaining, closed hashing – linear probing, quadratic probing, double hashing, random probing, rehashing, extendible hashing

Module:7	Tree and Graph	7 hours

Implementation of tree, binary tree traversals, expression tree, binary search tree, AVL tree Graphs, Graph traversals, and shortest path algorithms-Dijkstra‘s algorithm

Module:8	Contemporary issues	2 hours
	Total Lecture hours:	**45 hours**

Text Book(s)	
1.	Mark Allen Weiss, –Data structures and algorithm analysis in C‖, 2nd edition, Pearson education, 2013.
Reference Books	
1. 2. 3. 4.	Debasis Samanta, –Classic data structures‖, PHI, 2nd edition, 2014. Seymour Lipschutz –Data Structures by Schaum Series‖ 2nd edition, TMH, 2013. Adam Drozdek, –Data structures and algorithms in C++‖, Cengage learning, 4th edition, 2015. Michael Goodrich, Roberto Tamassta, Michael H.GoldWasser –Data structures and algorithms in Java‖ 6th edition, 2014.

List of Challenging Experiments (Indicative)		
1.	Implement stack and use it to convert infix to postfix expression	
2.	Evaluate postfix expression	
3.	Implement Towers of Hanoi problem	
4.	Implement Queue and Circular Queue	
5.	Implement singly and doubly linked lists	
6.	Implement Circular Singly Linked list	
7.	Represent a polynomial as a linked list and write functions for polynomial addition.	
8.	Implement Insertion, Bubble, and selection sorts	
9.	Implement heap, merge, quick and radix sorts	
10.	Implement Binary and Linear search	
11.	Implement a Binary tree. Produce its pre-order, in-order, and post-order traversals.	
12.	Implement binary search tree insertion and deletion.	
13	Implement hashing techniques	
14	Perform Graph traversal	
15	Implement Dijkstra's algorithm	
	STACK ADT 1. Students of a Programming class arrive to submit assignments. Their register numbers are stored in a LIFO list in the order in which the assignments are submitted. Write a program using array to display the register number of the ten students who submitted first.	

Register number of the ten students who submitted first will be at the bottom of the LIFO list. Hence pop out the required number of elements from the top so as to retrieve and display the first 10 students.

2. To facilitate a thorough net surfing, any web browser has back and forward buttons that allow the user to move backward and forward through a series of web pages. To allow the user to move both forward and backward two stacks are employed. When the user presses the back button, the link to the current web page is stored on a separate stack for the forward button. As the user moves backward through a series of previous pages, the link to each page is moved in turn from the back to the forward stack.

When the user presses the forward button, the action is the reverse of the back button. Now the item from the forward stack is popped, and becomes the current web page. The previous web page is pushed on the back stack. Simulate the functioning of these buttons using array implementation of

Stack. Also provide options for displaying the contents of both the stacks whenever required.

3. Design a program to employ a stack for balancing symbols such as parentheses, flower braces and square brackets, in the code snippet given below.

```
For(i=0;i<n;i++)

{

if(i<5)

{  z[i]=x[i]+y[i];

      p=(((a+b)*c)+(d/(e+f)*g);

 }
```

Ensure that your program works for any arbitrary expression.

4. Most of the bugs in scientific and engineering applications are due to improper usage of precedence order in arithmetic expressions. Thus it is necessary to use an appropriate notation that would evaluate the expression without taking into account the precedence order and parenthesis.

a) Write a program to convert the given arithmetic expression into

i) Reverse Polish notation

	ii) Polish notation b) Evaluate the above notations with necessary input. 5. Some priests are given three poles and a stack of 4 gold disks, each disk a little smaller than the one beneath it. Their assignment is to transfer all 4 disks from one of the 3 pole to another with 2 important constraints. They can move only one disk at a time, and they can never place a larger disk on top of a smaller one. Design a recursive program for the above Towers of Hanoi puzzle using stack. **QUEUE ADT:** 6. In a theme park, the Roller-Coaster ride is started only when a good number of riders line up in the counter (say 20 members). When the ride proceeds with these 20 members, a new set of riders will line up in the counter. This keeps continuing. Implement the above scenario of lining up and processing using arrays with Queue ADT. 7. When burning a DVD it is essential that the laser beam burning pits onto the surface is constantly fed with data, otherwise the DVD fails. Most leading DVD burn applications make use of a circular buffer to stream data from the hard disk onto the DVD. The first part, the _writing process‘ fills up a circular buffer with data, then the _burning process‘ begins to read from the buffer as the laser beam burns pits onto the surface of the DVD. If the buffer starts to become empty, the application should continue filling up the emptied space in the buffer with new data from the disk. Implement this scenario using Circular Queue. 8. a) There is a garage where the access road can accommodate any number of trucks at one time. The garage is built in such a way that only the last truck entered can be moved out. Each of the trucks is identified by a positive integer (a truck_id). Implement dynamically to handle truck moves, allowing for the following commands: 1) On_road (truck_id); ii) Enter_garage (truck_ id); 1) Exit_garage (truck_id); iv) Show_trucks (garage or road); If an attempt is made to get a truck out which is not the closest to the garage entry, the error message –Truck x cannot be moved‖ should be displayed. 1) For the aforementioned scenario, assume now a circular road and two entries: one for entry, another for exit. Trucks can get out only in the orderthey got in. Write a program dynamically to handle truck moves allowingfor the following commands i) Enter garage (truck name)	

ii) Exit garage (truck name)

iii) Show trucks

LIST ADT

9. Imagine an effective dynamic structure for storing polynomials. Write operations for addition, subtraction, and multiplication of polynomials.

I/O description. Input:

$p1=3x^7+5x^6+22.5x^5+0.35x^2$

$p2=0.25x^3+0.33x^2-0.01$

10. Given two sorted lists L1 and L2 write a program to merge the two lists in sorted order after eliminating duplicates.

11. Write a program to maintain the records of students in an effective dynamic structure. Search a particular record based on the roll number and display the previous and next values of that node with time complexity of O(1).

12. **Assume FLAMES** game that tests for relationship has to be implemented using a dynamic structure. The letters in the FLAMES stand for Friends, Love, Affection, Marriage, Enmity and Sister. Initially store the individual letters of the word _flames' in the nodes of the dynamic structure. Given the count of the number of uncommon letters in the two names _n', write a program to delete every nth node in it, till it is left with a single node. If the end of the dynamic structure is reached while counting, resume the counting from the beginning. Display the letter that still remains and the corresponding relationship

Eg., If Ajay and Jack are the two names, there are 4 uncommon letters in these. So delete 4^{th} node in the first iteration and for the next iteration start counting from the node following the deleted node.

SORTING AND SEARCHING

13. Assume in the Regional Passport Office, a multitude of applicants arrive each day for passport renewal. A list is maintained in the database to store the renewed passports arranged in the increased order of passport ID. The list already would contain there cords renewed till the previous day. Apply Insertion sort technique to place the current day's records in the list.

Later the office personnel wish to sort the records based on the date of renewal so as to know the count of renewals done each day. Taking into

	consideration the fact that each record has several fields (around 25 fields), follow Selection sort logic to implement the same. 14. Implement a comparison based sorting algorithm which is not in-place to sort the following strings. Best, true, hill, dove, van, good, egg, lap 15. Write a program to implement Bubble sort, Heap sort and Quick sort techniques to arrange the following sequence of elements in descending order. 9, -4, 5, 8,-3, 7, 0, 4, 1, 2. Display the count of number of comparisons and swaps made in each method. Apply the same sorting techniques for sorting a large data set [Randomly generate 5000 integers within the range -50000 to 50000 to build the data set]. From your observation and analysis, determine the best sorting technique for working with large numbers.	
	Total Laboratory Hours	30 hours

Recommended by Board of Studies	4-12-2015		
Approved by Academic Council	No. 39th	Date	17-12-2015

SWE2002	Computer Networks	L	T	P	J	C
		3	0	2	0	4
Pre-requisite	CSE1001	Syllabus version				
		v.1.0				

Course Objectives:

1. To learn the principles of computer networks including the Internet protocol stack and the OSI model.
2. To understand the working of LAN, WAN, MAN.
3. To analyses Error Control and Flow Control Protocols, Routing and Congestion Control Algorithms, Network Management and Performance.

Expected Course Outcome:

1. Understand the principles of layered architecture, Internet protocol stack, and the OSI model
2. Examine the internal mechanisms of packet switching and circuit switching, ATM and frame Relay technologies.
3. Understand Medium access control protocols LAN technologies,and Error control mechanisms.
4. To understand internetworking concepts and analyze the Network layer protocols
5. To analyze Routing algorithms and understand interconnecting devices.
6. To understand the transport layer protocols, and identify various flow control mechanisms.
7. Understand and use congestion control mechanisms.
8. Understand computer networks industry best practices related

Module:1	Overview of computer and communication networks:	6 hours

Networking principles; Network protocol-syntax, semantics, and timing; Layered protocol Stack; Protocol suites-OSI and TCP/IP. Network Standards and standardization bodies.

Module:2	Switched Communication Networks:	6 hours

LAN topologies. Switching – Circuit Switching-X.25 Network and Frame Relay, Packet Switching-Virtual and Datagram switching and Cell switching-ATM architecture, ATM layers, ATM cell format, Multiple access.

Module:3	Data link control:	6 hours

Link layer services, Framing, Medium Access-CSMA and CSMA/CD, LAN technologies-Ethernet, Gigabit Ethernet and Token Ring, Error Detection and Correction.

Module:4	Internetworking:	6 hours

Internet protocols – Ipv4 and Ipv6, ICMP, ARP, DHCP. Logical addressing-Ipv4 Subnetting and Classless addressing (CIDR) and Ipv6 addresses. Transition from Ipv4 to Ipv6. Internet header checksum, Networking utilities commands.

Module:5	**Internet Routing:**	**6 hours**
Routing algorithms- Distance vector and Link state routing, Internet Routing protocols-RIP, OSPF and BGP. Basic concepts of hubs, bridges, switches, gateways, and routers.		
Module:6	**Transport protocols:**	**6 hours**
Transport Protocols-UDP,-Reliable byte stream (TCP)-Connection Management, Flow control and Retransmission, TCP States, Transport header checksum, TCP and UDP client/server programming.		
Module:7	**Congestion control mechanisms:**	**7 hours**
TCP Congestion Control-Slow Start, Congestion avoidance, Fast retransmit and Fast Recovery. Congestion Detection Methods-Random Early Detection and Explicit Congestion Notification (ECN).		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)	
1.	W. Stallings, Data and Computer Communications, 10th Edition, Pearson Education, 2013.
Reference Books	
1.	Behrouz A Forouzan, Data Communications and Networking, 5th Edition, Tata Mc-grawhill, 2013.
2.	Andrew S. Tanenbaum, David J. Wetheral, Computer networks, 5th Edition, Pearson,2012.
3.	Nader F. Mir, Computer and Communication Networks, 2nd Edition,Pearson,PHI,2015
4.	Elliotte Rusty Harold,Java Network Programming, 4th Edition,O'Reilly Media,2013

List of Challenging Experiments (Indicative)		
1.	Using TCP sockets, write a simple Java program to display the current date and time.	
2.	Write a program to implement a simple message transfer from client to server process using TCP sockets.	
3.	Write a TCP socket program to display, in client window, the sum of random numbers generated by the server.	
4.	Write a program to implement a chat server and client in java using TCP sockets.	
5.	The message entered in the client is sent to the server and the server encodes the message and returns it to the client. Encoding is done by replacing a character by the character next to it (i.e. a as b, b as c ...z as a). This process is done using the TCP/IP protocol. Write a Java	

	program for the above.	
6.	Write a program to implement a simple message transfer from client to server process using UDP sockets.	
	SAMPLE CHALLENGING EXERCISES 1. There are 20PC‘s in your network. Five PC‘s are connected to one Ethernet hub, and five PC‘s are connected to another hub. Each hub is connected to separate switch and both the switches are connected to a separate router. The routers are connected via an Ethernet bridge. The remaining 10 PC‘s are connected directly to one of the two switches. How many Ethernet segments are there? Implement this scenario using cisco packet tracer. 2. Two PC‘s are located in adjacent rooms and a third PC is in a building 300 yards away. Explain how you could connect the three PC‘s in a single network. Implement this scenario using cisco packet tracer.. 3. In CRC error correction scheme, choose pattern 1101 and data 100100. Write a code to encode the given data. 4. There is trouble ticket raised by users of an organization that their files are not getting uploaded in ftp server. Measure the performance between the ftp server and client and diagnose using iperf tool. 5. A company needs is granted the site address 201.70.64.0. The company needs six subnets. Design the subnets using cisco packet tracer. 6. In an Ipv4 packet the value of header length is 1000 in binary. Write a code to find, how many bytes of options are being carried by this packet? 7. Write a code to implement border gateway protocol (BGP). 8. Implement a TCP/IP socket based ATM System. Make the server to maintain the customer details (name, card no, pin and balance). When a client wants to withdraw amount, validate his login with card no & pin, display a welcome message and perform the withdraw operation if he is having sufficient balance or display a warning message. 9. Write a UDP based server code to get the date of birth of the client and calculate the age as on today. Client has to enter year, month and day of birth. For example, if the date of birth of a user is 1/07/2001 then his age is 14 years 0 months and 17 days if today’s date is 18/07/2015. Get today’sdate from the server. 10. A reputed organization has two branches in Vellore. In one of the branch office a new manager has been appointed. The Senior Manager from the main office has to send the important records to the branch	

	office. Implement a client server model to accomplish this. 1. The finance office of VIT wishes to make the transactions more secured. If you are a programmer how you will implement a system to validate the login credentials obtained from the user thereby denying the access to unauthorized users. Establish a wired network running many applications level services and measure the performance of same. Establish a wireless network running many applications level services and measure the performance of same. Compare the performance of above two scenarios and list out the challenges.	
	Total Laboratory Hours	30 hours

Recommended by Board of Studies	5-3-2016		
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE 2003	Requirements Engineering and Management	L	T	P	J	C
		2	0	0	4	3
Pre-requisite	SWE1701	Syllabus version				
		v 1.20				

Course Objectives:

1. To understand the need of requirements for engineering large scale systems
2. To specify functional requirements and non-functional requirements
3. To analysis given problem-scenarios

Expected Course Outcome:

1. Understand the importance of software process models and requirements management
2. Understand business modeling and systems engineering.
3. Recognize the various strategies of requirement elicitation process and appreciate the challenges of requirement elicitation.
4. Develop vison and scope document
5. Specify functional requirements, nonfunctional requirement and design constraints
6. Appreciate the usage of requirement management tools

Module:1	Requirements Management and Problem Analysis	4 hours

The Requirements Problem – The Root Causes of Project Success and Failure. Introduction to Requirements Management – The Road Map. The Requirements and the Software Lifecycle – Traditional Software Process models – The Iterative approach, Requirements in the Iterative Approach. The five steps in Problem Analysis – Discussion on a Case Study.

Module:2	Business Modeling and Systems Engineering	4 hours

Business Modeling – The Purpose of Business Modeling, Using Software Engineering Techniques for Business Modeling, From the Business Model to the Systems Model. Systems Engineering of Software intensive systems –Requirements Allocation in Systems Engineering – The Case study in System Engineering.

Module:3	Requirements Gathering Techniques	8 hours

The Challenge of Requirements Elicitation – Barriers to Elicitation – The Features of a Product or System – Stakeholder and User Needs, Features. Interviewing – Requirements Workshops – Brainstorming and Idea Reduction – Storyboarding.- Technical Methods for Specifying Requirements- Finite State Machines – Decision Tables and Decision Trees – Activity Diagrams-

Entity-Relationship Models.		
Module:4	**Defining the System**	**4 hours**
A Use case Primer – Organizing Requirements Information – Organizing Requirements of Complex Hardware and Software Systems, Organizing Requirements for Product Families. The Vision Document. Product Management – The Role of Product Champion – Primary Activities fora Product Manager – Supporting Activities .Establishing Project Scope – The Problem of Project Scope – The Requirements Baseline Setting.		
Module:5	**Refining the System Definition**	**8 hours**
Software Requirements – Refining the Use Cases – How Use Cases Evolve- The Scope of Use case- Extending Use Case- Developing the Supplementary Specification.- Building the Right System- From Use Cases to Implementation – Mapping Requirements to Design and code – FromUse Cases to Test Cases- Tracing Requirements – The Traceability Relationship – Using Traceability Tool.		
Module:6	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**30 hours**

Text Book(s)

1. Dean Leffingwell, Don Widrig, "Managing Software Requirements: A Use Case Approach", Pearson Higher Education, 2nd Edition, 2013.

Reference Books

1. Klaus Pohl, ‖Requirements Engineering – Fundamentals, Principles and Techniques‖, Springer – Verlag Berlin Heidelberg 2010.
2. Karl Wiegers, Joy Beatty, "Software Requirements", Addison - Wesley Professional,3rd edition, 2013.
3. Suzanne Robertson, James Robertson, "Mastering the Requirements Process: Getting Requirements Right", Addison – Wesley Professional; 3rd edition, 2012.
4. Aurum, Aybüke, Wohlin, Claes (Editors), "Engineering and Managing Software Requirements", Springer – Verlag Berlin Heidelberg, 2005.
5. Ian Sommerville, Pete Sawyer, "Requirements Engineering: A Good Practice Guide," Wiley, 2009.

Recommended by Board of Studies	4-12-2015		
Approved by Academic Council	No. 39th	Date	17-12-2015

SWE2004	Software Architecture and Design	L	T	P	J	C
		2	0	0	4	3
Pre-requisite	**SWE1701**	**Syllabus version**				
		v. 1.0				

Course Objectives:

1. To Understand Software architecture and design principles
2. To analyze the software requirements and evaluating the designs
3. To apply various techniques and methods involved in creating model of a Software design.
4. To use software architectural styles based on the design viewpoints, design rules and user interfaces.

Expected Course Outcome:

1. Relate design process principles to software quality factors.
2. Understand the software design strategies, develop design thinking capability.
3. Apply different types of systems analysis techniques and software design strategies
4. Distinguish different types of software architectural styles
5. Formulate user interface design rules and describes shared information system with design principles, standards and guidelines.
6. Evaluate and implement different types of design patterns based on the requirement and functionality
7. Summarizing different types of software design issues and software design tools
8. Exemplify software design techniques and design patterns to validate design

Module:1	**Design fundamentals**	**5 hours**

Nature of Design process objectives, Building Modules, Constructs, Design qualities, assessing the design, Design viewpoints for software. Design practices-Analysis on design requirements and designing with quality factors, coupling, cohesion and cognitive dimensions, measure quality attributes and assessment – Case studies.

Module:2	**Design strategies and Methodologies**	**5 hours**

Design strategies Top down and bottom up, Organizational methods and design. Jackson Structural programming, Jackson system development.

Module:3	**Design Models**	**5 hours**

Object-based design and Structured System Analysis and Structured design method Traditional approach to design-SADT organizational design practices-SSADM and design for real time systems. Case study: Analysis on ‒Home safety security systems‖ by applying SSADMObject-based design and Structured System Analysis and Structured design method Traditional approach to design-SADT organizational design practices-SSADM and design for real time systems. Case study: Analysis on ‒Home safety security systems‖ by applying SSADM

Module:4	**Software Architecture**	**7 hours**

Introduction- Software Architecture- Definition Prospects- State of Art-Architectural Styles-Pipes and Filters-Layered Systems-Repositories-Process Control, Other familiar Architecture-Heterogeneous Architectures. Case studies. – **Architecture design-** Introduction Shared Information System, Architecture Structures for Shared Information Systems. Architecture design guidance-User Interface Architecture. Case study: Design of User Interface Design

Module:5	Software Architecture patterns	6 hours
Introduction to design pattern Architectural design and Mapping–Description of various Architectural design patterns. – **Emerging Trends in Architecture and Design –** Tools for Architectural design Exploiting style in architectural design – Architectural Interconnection. Case studies – Architecture and Design tools.		
Module:6	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**30 hours**

Text Book(s)

1.	DavidBudgen," SoftwareDesign",AddisonWesley,Pearson Education2ndEdition 2012

Reference Books

1 Hong Zhu, ‒Software Design Methodology From Principles to Architectural Styles‖, Elsevier,2011.
2 R.S.Pressman, "Software Engineering", Fifth Edition, McGraw Hill Inc., 2015.
3 MaryShawDavidGarlan,"SoftwareArchitecturalPerspectivesonanemergingdiscipline", EEE, PHI, 2011.

Recommended by Board of Studies	5-3-2016		
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE2005	Software Testing	L	T	P	J	C
		3	0	0	4	4
Pre-requisite	**SWE1701**	**Syllabus version**				
		v. 1.0				

Course Objectives:

1. To learn fundamental concepts in software testing
2. To indentify various software testing issues and solutions in software unit test; integration, regression, and system testing.
3. Test project, design test cases and data.
4. To plan and excute a testing project for use modern software testing tools to support software testing projects.

Expected Course Outcome:

1. Apply software testing knowledge and engineering methods.
2. Examine and solve various functionality problems by designing and selecting testing models and methods.
3. Examine and slove various program logic or structure problems, by designing and selecting testing models and methods.
4. Develop construct the complementary techniques to dynamic testing for improving the software quality
5. Design and experiment a software test process for a software project
6. Interpret and review the contemporary issues in software testing, such as component-based software testing problems.
7. Apply debugging process and techniques for software engineering problems.
8. Use and demonstrate software testing methods and modern software testing tools for their testing projects.

Module:1	**Introduction**	**6 hours**

Introduction –Evolution of Software testing- Myths and Facts-Goals-Psychology –Software Testing Models- Different Schools of software testing-Software testing Life cycle – Testing methodology.

Module:2	**Black box testing strategies**	**5 hours**

Black-Box Testing Techniques- Equivalent partitioning-Boundary Value Analysis (BVA)- State Transition Testing-Decision table based Testing – Cause-Effect Graphing Based Testing – Error Guessing

Module:3	**White box testing strategies**	**7 hours**

White-Box Testing Techniques- Logic Coverage criteria-Basic path testing-Graph matrices-Loop testing-Data flow testing-Mutation testing

Module:4	**Verification and Validation Testing**	**6 hours**

Inspection-Structured walkthrough- technical reviews-Unit –Integration –System –Acceptance testing-System testing

Module:5	**Maintenance and Management**.	**6 hours**
Regression testing –objectives- Types-Test organization –Structure of test group_ Test planning-Test Design and Design specifications		
Module:6	**Object Oriented Testing and Web Based Testing**	**7 hours**
OO Testing basic-OOT testing –Web based system-Evolution –challenges-Quality aspects –web engineering		
Module:7	**Debugging and Test Maturity models**	**6 hours**
Debugging- Process – Techniques-Correction of Bugs – debuggers.-Need for process maturity – Measurement and Improvement of test process-Test process maturity models		
Module:8	**Contemporary issues:** Applications of Software Testing in industry	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1.	NareshChauhan –Software Testing Principles and Practices‖,Oxford University Press, 2010

Reference Books

1	William E- Perry –Effective methods for software testing‖ –Wiley publications -2006.
2	Ilene Burnstein, "Practical Software Testing", Springer Verlag International Edition, Springer (India) Pvt Ltd – (Indian reprint edition 2013)

Recommended by Board of Studies	5-3-2016		
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE2006	Software Project Management	L	T	P	J	C
		2	0	0	4	3
Pre-requisite	**SWE1701**	**Syllabus version**				
		v.1.0				

Course Objectives:

1. To characterize Software projects and understand project management activities
2. To gain knowledge about software estimation techniques and management
3. To monitor and control software projects and to manage people as well as build teams.

Expected Course Outcome:

1. To understand Project Management activities and to identify types of software projects.
2. Select Software projects using Cost Benefit Analysis (CBA).
3. Apply critical path method CPM to estimate the project duration and shorten project duration
4. Develop activity network to use PERT and to manage project risks
5. Identify Visualization techniques for software project planning and apply Earned Value Analysis to know the status of the Project.
6. Understand contracts and managing steps for Contracts
7. Assess and select people for software projects
8. Develop an estimate for a given software project scenario

Module:1	INTRODUCTION TO SOFTWARE PROJECT MANAGEMENT	3 hours

Project Definition – Contract Management – Activities covered By Software Project Management – Overview of Project Planning – Stepwise Project Planning.

Module:2	PROJECT EVALUATION AND ACTIVITY PLANNING	8 hours

Strategic Assessment – Technical Assessment – Cost Benefit Analysis –Cash Flow Forecasting – Cost Benefit Evaluation Techniques – Risk Evaluation – Objectives – Project Schedule – Sequencing and Scheduling Activities –Network Planning Models – Forward Pass – Backward Pass – Activity Float – Shortening Project Duration – Activity on Arrow Networks

Module:3	RISK MANAGEMENT	4 hours

Nature Of Risk – Types Of Risk – Managing Risk – Hazard Identification – Hazard Analysis – Risk Planning And Control

Module:4	MONITORING AND CONTROL	7 hours

Creating Framework – Collecting The Data – Visualizing Progress – Cost Monitoring – Earned Value – Prioritizing Monitoring – Getting Project Back To Target – Change Control – Managing Contracts – Introduction – Types Of Contract – Stages In Contract Placement – Typical Terms Of A Contract – Contract Management – Acceptance.

Module:5	MANAGING PEOPLE AND ORGANIZING TEAMS	6 hours

Introduction – Understanding Behavior – Organizational Behaviour: A Background – Selecting The Right Person For The Job – Instruction In The Best Methods – Motivation – The Oldham – Hackman Job Characteristics Model – Working In Groups – Becoming A Team –Decision Making – Leadership – Organizational Structures – Stress –Health And Safety – Case Studies.

Module:6	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**30 hours**

Text Book(s)	
1.	Mike Cotterell, Bob Hughes, Rajib Mall – Software Project Management – Tata McGraw-Hill, Fifth Edition – 2011.
Reference Books	
1.	Ramesh Gopalaswamy – Managing Global Projects – Tata McGraw Hill – First Edition, 2006.
2.	Greg Horine-Project Management Absolute Beginner's Guide, 3/E- Que Publishing ,2012.

Recommended by Board of Studies	5-3-2016		
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE2007	Software Construction and Maintenance	L	T	P	J	C
		2	0	0	4	3
Pre-requisite	SWE1701	Syllabus version				
		v.1.0				

Course Objectives:

1. To construct a software using any known programming language
2. To gain knowledge about best practices in software construction
3. To recognize the role of maintenance in software development.
4. To understand the issues related to out sourcing software projects and work on a software maintenance project.

Expected Course Outcome:

1. Apply the fundamentals of software construction and appreciate the challenges in software construction.
2. Interpret key practical construction considerations such as design, languages, coding, testing, quality and reuse.
3. Understand and recognize the importance of modern construction technologies
4. Learn about Construction Tools including development environments, GUI Builders etc.
5. Comprehend software evolution and birds eye view of software maintenance.
6. Appreciate the value of problem resolution in maintenance
7. Understand about distribution of fixes, methods, tools, composition and people issues.
8. Appreciate the value of software construction and maintenance and challenges faced in software industry

Module:1	Software Construction Fundamentals and Managing Construction	4 hours

Software Construction Fundamentals: Minimizing Complexity; Anticipating Change; Constructing for Verification; Reuse; Standards in Construction, Managing Construction: Construction in Life Cycle Models; Construction Planning; Construction Measurement

Module:2	Practical Considerations	4 hours

Construction Design; Construction Languages; Coding; Construction Testing; Construction for Reuse; Construction with Reuse; Construction Quality; Integration

Module:3	Construction Technologies	5 hours

API Design and Use, Object-Oriented Runtime Issues, Parameterization and Generics; Assertions, Design by Contract, and Defensive Programming; Error Handling, Exception Handling, and Fault Tolerance; Executable Models; State-Based and Table-Driven Construction Techniques; Runtime Configuration and Internationalization – Development Environments; GUI Builders; Unit Testing
Tools; Profiling, Performance Analysis and Slicing Tools

Module:4	Software Maintenance Basics	6 hours
Software Maintenance; customer‘s View point; Economic of Maintenance; A Bird‘s Eye view of Maintenance; Different type of software products; An Overview of corrective Maintenance; Other forms of Maintenance; Adaptive Maintenance; Enhancement Requests; Maintenance Processes; Customer side preliminary Activities; Skill sets needed for the various Roles During Problem Reporting.		
Module:5	**Problem Resolution & Fix Distribution**	**9 hours**
Problem Resolution: High Level Overview of Activities in problem Resolution; Categorizing the problem; Identifying the Right Developer for fixing the problem; Reproducing the problem; Scheduling for release. **Fix Distribution:** High Level Overview of Activities in problem Resolution; Categorizing the problem; Identifying the Right Developer for fixing the problem; Reproducing the problem; Scheduling for release.		
Module:6	**Contemporary issues**	**2 hours** **-**
	Total Lecture hours:	**30 hours**

Text Books

1. McConnell, Steve, Code complete: A practical handbook of software construction, 2nd Edition, Microsoft Press, 2012.
2. Gopalaswamy Ramesh and Ramesh Bhattiprolu, Software Maintenance – Effective Practices forGeographically Distributed Environments, Tata McGraw-Hill Education, 2012.

Reference Books

1. A. Hunt and D. Thomas, The Pragmatic Programmer – from journey man to master, Addison-Wesley, 2010.
2. B.W. Kernighan and R. Pike, The Practice of Programming, Pearson Eductaion India, 2012.
3. SWEBOK V3.0, Guide to the Software Engineering Body of Knowledge, A Project of the IEEE Computer Society, 2014.

Recommended by Board of Studies	5-3-2016		
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE3001	Operating Systems	L	T	P	J	C
		3	0	2	0	4
Pre-requisite	SWE2001	Syllabus version				
		v. 1.0				

Course Objectives:

1. To understand the services provided by and the design of an operating system.
2. To understand the structure and organization of the file system
3. To understand principles of process management anddifferent approaches to memory management.

Expected Course Outcome:

1. Understand principles and modules of operating system.
2. Understand key mechanisms in design of operating system modules
3. Compare various processor scheduling algorithms.
4. Develop algorithmic solutions to process synchronization problems.
5. Understand CPU scheduling for distributed operating systems
6. Understand the mechanisms adopted for file sharing in distributed Applications Identify components involved in designing a contemporary OS
7. Identify the components involved in designing a contemporary OS

Module:1	Introduction	6 hours

Computer-System Organization, Computer-System Architecture, Operating-System Structure, Operating-System Operations, Operating-System Services, User and Operating-System Interface, System Calls, Operating-System Generation, System Boot.

Module:2	Processes	6 hours

Process Concept, Process Scheduling, Operations on Processes, Inter-process Communication, Threads- Overview, Multithreading Models, Thread Libraries, Implicit Threading, Threading Issues

Module:3	Process Synchronization	6 hours

Background, The Critical-Section Problem, Peterson‘s Solution, Synchronization Hardware, Mutex Locks, Semaphores, Classic Problems of Synchronization, Monitors, Synchronization Example

Module:4	CPU Scheduling	6 hours

Basic Concepts, Scheduling Criteria, Scheduling Algorithms, Threads, Multiple-Processor Scheduling, Deadlocks- System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock.

Module:5	Memory Management	6 hours

Background, Swapping, Contiguous Memory Allocation, Segmentation, Paging, structure of the Page Table.

Module:6	Virtual Memory	6 hours

Background, Demand Paging, Page Replacement, Allocation of Frames, Thrashing, Memory-Mapped Files, Allocating Kernel Memory

Module:7	**Mass-Storage Structure**	**7 hours**
Overview, Disk Structure, Disk Scheduling, Disk Management, Swap-Space Management, RAID Structure, File-System Interface- File Concept, Access Methods, Directory and Disk Structure, File-System Mounting, File Sharing, Directory Implementation, Allocation Methods.		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1. A.Silberschatz, P.B. Galvin & G. Gagne, Operating system concepts, Ninth Edition, John Wiley, 2013

Reference Books

1. W. Stallings, Operating systems-Internals and Design Principles, Seventh Edition , Prentice-Hall,2012
2. Tanenbaum, Modern Operating Systems, Third Edition, PrenticeHall,2015

List of Challenging Experiments (Indicative)

1.	Process scheduling mechanism	
2.	Readers – Writers Problem	
3.	Dining Philospher‘s Problem	
4.	Deadlock – Banker‘s Algorithm	
5.	Page Replacement Algorithm Implementation	
	Total Laboratory Hours	30 hours

Recommended by Board of Studies	5-3-2016		
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE3002	Information & Systems Security	L	T	P	J	C
		3	0	0	4	4
Pre-requisite	**SWE2002**	**Syllabus version**				
		v.1.0				

Course Objectives:

1. To learn principles of cryptography, network and information security.
2. To comprehend mathematical foundations of cryptography
3. To introduce the practices of cryptography and network security along with its applications
4. To use the information sources

Expected Course Outcomes:

1. Identify the challenges of security attacks
2. Understand the elementary cryptography based on symmetric and public-key encryption techniques
3. Understand public Key Crypto Systems models, RSA algorithm, Diffie-Hellman key exchange
4. Apply Cryptographic hash functions SHA-512, MAC requirements, security, HMAC, Digital signatures
5. To generate the key distributions using symmetric and asymmetric encryptions
6. Enumerate malicious software, viruses and counter measures
7. Understand Operating Systems & Data base Security issues and control methods
8. Study Applications of Information & Systems Security in industry

Module:1	**Fundamentals of Security**	**6 hours**

Definitions & challenges of security, OSI security architecture, Attacks & services, Security policies, Access control structures.

Module:2	**Elementary Cryptography**	**6 hours**

Cryptography & cryptanalysis. Classical encryption techniques, Substitution techniques, Transposition techniques. Block ciphers, DES, AES structure.

Module:3	**Public Key Crypto Systems**	**6 hours**

Number theory fundamentals, Principles of pubic key crypto systems, RSA algorithm, Diffie-Hellman key exchange.

Module:4	**Authentication Protocols**	**6 hours**

Cryptographic hash functions, applications, requirements, SHA-512, MAC requirements, security, HMAC, Digital signatures.

Module:5	**Key Management & Distribution**	**6 hours**

Symmetric key distribution using symmetric and asymmetric encryptions, Distribution of public keys, PKI.

Module:6	**Program Security**	**6 hours**

Secure programs, Non malicious program errors, Types of malicious software, Viruses and counter measures, Bots, Rootkits, Targeted malicious code, Controls against program threats, Software security issues.

Module:7	**Operating Systems & Database Security**	**7 hours**
Protected objects and Methods of protection, Memory and Address protection, Control of access to general objects, Kernel flaws, File protection Mechanisms, Security requirements of databases, Sensitive data, Inference, Multilevel secure databases, Concurrency control and Multilevel security.		
Module:8	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)	
1.	William Stallings, Cryptography & Network Security- Principles and Practices, 6th Edition by Pearson Publishers, 2014.
Reference Books	
.1	William Stallings, Lawrie Brown, Computer Security: Principles and Practice, 3rd edition, 2014.
2.	Christof Paar & Jan Pelzl, Understanding Cryptography, Springer, 2010.
3	Charles P. Pfleeger, Security in Computing, 4th Edition, Pearson, 2009.

Recommended by Board of Studies		5-3-2016	
Approved by Academic Council	No. 40th	Date	18-3-2016

BIT1029	Basic Bioinformatics	L	T	P	J	C
		3	0	0	0	3
Pre-requisite	NONE	Syllabus version				
		v.1.0				

Course Objectives:

- The students would be able to understand and explain the fundamentals of Bio-informatics, Dynamic programming, searching algorithms, Evolutionary trees, DNA mapping, DNA sequencing and Gene predictions

Expected Course Outcome:

- Students will interpret relationships among living things and analyze and solve biological problems, from the molecular to ecosystem level using basic biological concepts, grounded in foundational theories.

Module:1	Introduction to Bioinformatics	6 Hours

Scope of Bioinformatics – Elementary commands and Protocols, ftp, telnet, http, Primer on information theory.

Module:2	Sequencing Alignment and Dynamic Programming	6 Hours

Introduction – Strings – Edit distance between two strings – string similarity local alignment gaps –Parametric sequence alignments – multiples alignment – common multiple alignment methods.

Module:3	Sequence Databases and Uses	6 Hours

Introduction to databases – database search – Algorithms issues in database search – sequence database search – FASTA – BLAST – Amino acid substitution matrices PAM AND BLOSSUM

Module:4	Evolutionary Trees and Phylogeny	6 Hours

Ultrasonic trees – parsimony – Ultrametric problem – Perfect phylogeny – Phylogenetic alignment –connection between multiple alignment and tree construction

Module:5	Special Topics in Bioinformatics	6 Hours

DNA Mapping and sequencing – Map alignment – Large scale sequencing and alignment – shotgun –DNA sequencing – sequence assembly – Gene predictions – Molecular predictions with DNA strings

Module:6	Strings and Evolutionary Trees	6 hours

Ultrametric trees and ultrametric distances – Additive-distance trees – Parsimony: charac ter-based evolutionary reconstruction – The centrality of the ultrametric problem – Maximum parsimony, Steiner trees, and perfect phylogeny Phylogenetic alignment, again – Connections
between multiple alignment and tree construction

Module:7	Matching DNA to protein	6 hours

Matching DNA to protein with frameshift errors – Gene prediction – Molecular computation:

computing with DNA strings			
Module:8	**Contemporary issues:**		3 hours
	Total Lecture hours:		**45 hours**
Text Book(s)			
1.	Dan Gusfield,(1997)"Algorithms On Strings Trees and Sequences", Cambridge University Press		
Reference Books			
1.	Westhead, "Instant notes – Bioinformatics", Viva Publishers. 2.Bergeron Bryan, "Bioinformatics Computing", Prentice Hall of India		
Recommended by Board of Studies		10-06-2015	
Approved by Academic Council		No. 37	Date 16-06-2015

Course Code	Information Security Analysis and Audit	L	T	P	J	C
CSE3501	Job Role: SSC/Q0901	2	0	2	4	4
Pre-requisite	NIL			Syllabus version		
				1.0		

Objective of the course

1. To introduce system security related incidents and insight on potential defenses, counter measures against common threat/vulnerabilities.
2. To provide the knowledge of installation, configuration and troubleshooting of information security devices.
3. To make students familiarize on the tools and common processes in information security audits and analysis of compromised systems.

Expected Outcome

After successfully completing the course the student should be able to

1. Contribute to managing information security
2. Co-ordinate responses to information security incidents
3. Contribute to information security audits
4. Support teams to prepare for and undergo information security audits
5. Maintain a healthy, safe and secure working environment
6. Provide data/information in standard formats
7. Develop knowledge, skills and competence in information security

1	**Information Security Fundamentals**	7 hours

Definitions & challenges of security, Attacks & services, Security policies, Security Controls, Access control structures, Cryptography, Deception, Ethical Hacking, Firewalls, Identify and Access Management (IdAM).

2	**System Security**	6 hours

System Vulnerabilities, Network Security Systems, System Security, System Security Tools, Web Security, Application Security, Intrusion Detection Systems,

3	**Information Security Management**	3 hours

Monitor systems and apply controls, security assessment using automated tools, backups of security devices, Performance Analysis, Root cause analysis and Resolution, Information Security Policies, Procedures, Standards and Guidelines

4	**Incident Management**	5 hours

Security requirements, Risk Management, Risk Assessment, Security incident management, third party security management, Incident Components, Roles.

5	**Incident Response**	4 hours

Incident Response Lifecycle, Record, classify and prioritize information security incidents using standard templates and tools, Responses to information security incidents, Vulnerability Assessment, Incident Analysis

6	**Conducting Security Audits**	3 hours

Common issues in audit tasks and how to deal with these, Different systems and structures that may need information security audits and how they operate, including: servers and storage devices, infrastructure and networks , application hosting and content management, communication routes such as messaging, Features, configuration and specifications of information security systems and devices and associated processes and architecture, Common audit techniques, Record and report audit tasks, Methods and techniques for testing compliance.

7	**Information Security Audit Preparation**	2 hours

Establish the nature and scope of information security audits, Roles and responsibilities, Identify the procedures/guidelines/checklists, Identify the requirements of information security, audits and prepare for audits in advance, Liaise with appropriate people to gather data/information required for information security audits.

8	**Self and Work Management**	2 hours

Establish and agree work requirements with appropriate people, Keep the immediate work area clean and tidy, utilize time effectively, Use resources correctly and efficiently, Treat confidential information

correctly, Work in line with organization's policies and procedures, Work within the limits of their job role.

	Total Lecture hours:	**30 hours**

Text Book(s)

1. William Stallings, Lawrie Brown, Computer Security: Principles and Practice, 3rd edition, 2014.
2. Nina Godbole, Information Systems Security: Security Management, Metrics, Frameworks and Best Practices, Wiley, 2017
3. Nina Godbole, Sunit Belapure, Cyber Security- Understanding cyber-crimes, computer forensics and legal perspectives, Wiley Publications, 2016
4. Andrew Vladimirov Michajlowski, Konstantin, Andrew A. Vladimirov, Konstantin V. Gavrilenko, Assessing Information Security: Strategies, Tactics, Logic and Framework, IT Governance Ltd, O'Reilly, 2010

Reference Books

1. Charles P. Pfleeger, Security in Computing, 4th Edition, Pearson, 2009.
2. Christopher J. Alberts, Audrey J. Dorofee , Managing Information Security Risks, Addison-Wesley Professional, 2004
3. Peter Zor, The Art of Computer Virus Research and Defense, Pearson Education Ltd, 2005
4. Lee Allen, Kevin Cardwell, Advanced Penetration Testing for Highly-Secured Environments – Second Edition, PACKT Publishers, 2016
5. Chuck Easttom , System Forensics Investigation and Response, Second Edition, Jones & Bartlett Learning, 2014
6. David Kennedy, Jim O'Gorman, Devon Kearns, and Mati Aharoni, Metasploit The Penetration Tester's
7. Guide, No Starch Press, 2014
8. Practical Malware Analysis by Michael Sikorski and Andrew Honig, No Starch Press, 2015
9. Ref Links:
   https://www.iso.org/isoiec-27001-information-security.html
   https://csrc.nist.gov/publications/detail/sp/800-55/rev-1/final
   https://www.sans.org/reading-room/whitepapers/threats/paper/34180
   https://www.sscnasscom.com/qualification-pack/SSC/Q0901/

List of Experiments (Indicative)

- Install and configure information security devices
- Security assessment of information security systems using automated tools.
- Vulnerability Identification and Prioritization
- Working with Exploits
- Password Cracking
- Web Application Security Configuration
- Patch Management
- Bypassing Antivirus Software
- Static Malware Analysis
- Dynamic Malware Analysis
- Penetration Testing
- MySQL SQL Injection
- Risk Assessment
- Information security incident Management
- Exhibit Security Analyst Role

Total Laboratory Hours	**30 hours**

Recommended by Board of Studies	05.02.2020		
Approved by Academic Council	58	Date	26.02.2020

Course Code	Information Security Management	L	T	P	J	C
CSE3502	Job Role: SSC/Q0901	2	0	2	4	4
Pre-requisite	NIL	Syllabus version				
		1.0				

Objective of the course

1. To introduce system security related incidents and insight on potential defenses, counter measures against common threat/vulnerabilities.
2. To provide the knowledge of installation, configuration and troubleshooting of information security devices.
3. To make students familiarize on the tools and common processes in information security audits and analysis of compromised systems.

Expected Outcome

After successfully completing the course the student should be able to

1. Contribute to managing information security
2. Co-ordinate responses to information security incidents
3. Contribute to information security audits
4. Support teams to prepare for and undergo information security audits
5. Maintain a healthy, safe and secure working environment
6. Provide data/information in standard formats
7. Develop knowledge, skills and competence in information security

1	**Information Security Devices**	5 hours
Identify And Access Management (IdAM), Networks (Wired And Wireless) Devices, Endpoints/Edge Devices, Storage Devices, Servers, Infrastructure Devices (e.g. Routers, Firewall Services) , Computer Assets, Servers And Storage Networks, Content management, IDS/IPS		
2	**Security Device Management**	6 hours
Different types of information security devices and their functions, Technical and configuration specifications, architecture concepts and design patterns and how these contribute to the security of design and devices.		
3	**Device Configuration**	5 hours
Common issues in installing or configuring information security devices, Methods to resolve these issues, Methods of testing installed/configured information security devices,		
4	**Information Security Audit Preparation**	5 hours
Establish the nature and scope of information security audits, Roles and responsibilities, Identify the procedures/guidelines/checklists, Identify the requirements of information security, audits and prepare for audits in advance, Liaise with appropriate people to gather data/information required for information security audits. **Security Audit Review -** Organize data/information required for information security audits using standard templates and tools, Audit tasks, Reviews, Comply with the organization's policies, standards, procedures, guidelines and checklists, Disaster Recovery Plan		
5	**Team Work and Communication**	2 hours
Communicate with colleagues clearly, concisely and accurately , Work with colleagues to integrate their work effectively, Pass on essential information to colleagues in line with organizational requirements, Identify any problems they have working with colleagues and take the initiative to solve these problems, Follow the organization's policies and procedures for working with colleagues		
6	**Managing Health and Safety**	2 hours
Comply with organization's current health, safety and security policies and procedures, Report any identified breaches in health, safety, and Security policies and procedures, Identify, report and correct any hazards, Organization's emergency procedures, Identify and recommend opportunities for improving health, safety, and security.		
7	**Data and Information Management**	3 hours
Fetching the data/information from reliable sources, Checking that the data/information is accurate, complete and up-to-date, Rule-based analysis of the data/information, Insert the data/information into the agreed		

formats, Reporting unresolved anomalies in the data/information.		
8	**Learning and Self Development**	2 hours
Identify accurately the knowledge and skills needed, Current level of knowledge, skills and competence and any learning and development needs, Plan of learning and development activities to address learning needs, Feedback from appropriate people, Review of knowledge, skills and competence regularly and appropriate action taken		
	Total Lecture hours:	**30 hours**

Text Book(s)

1. Information Systems Security: Security Management, Metrics, Frameworks and Best Practices, Nina Godbole, Wiley, 2017
2. Rhodes-Ousley, Mark. Information Security: The Complete Reference, Second Edition, . Information Security Management: Concepts and Practice. New York, McGraw-Hill, 2013.
3. Christopher J. Alberts, Audrey J. Dorofee , Managing Information Security Risks, Addison-Wesley Professional, 2004

Reference Books

1. Andrew Vladimirov Michajlowski, Konstantin, Andrew A. Vladimirov, Konstantin V. Gavrilenko, Assessing Information Security: Strategies, Tactics, Logic and Framework, IT Governance Ltd, O'Reilly 2010
2. Christopher J. Alberts, Audrey J. Dorofee , Managing Information Security Risks, Addison-Wesley Professional, 2004
3. Chuck Easttom , System Forensics Investigation and Response, Second Edition, Jones & Bartlett Learning, 2014
4. David Kennedy, Jim O'Gorman, Devon Kearns, and Mati Aharoni, Metasploit The Penetration Tester's
5. Guide, No Starch Press, 2014

Ref Links:
https://www.iso.org/isoiec-27001-information-security.html
https://www.sans.org/reading-room/whitepapers/threats/paper/34180
https://csrc.nist.gov/publications/detail/sp/800-40/version-20/archive/2005-11-16
https://www.sscnasscom.com/qualification-pack/SSC/Q0901/

List of Experiments (Indicative)

1.
- Install and configure information security devices
- Penetration Testing
- MySQL SQL Injection
- Information security incident Management
- Intrusion Detection/Prevention
- Port Redirection and Tunneling
- Exploring the Metasploit Framework
- Working with Commercial Tools like HP Web Inspect and IBM AppScan etc.,
- Explore Open Source tools like sqlmap, Nessus, Nmap etc
- Documentation with Security Templates from ITIL
- Carry out backups of security devices and applications in line with information security policies, procedures and guidelines
- Information security audit Tasks - Procedures/guidelines/checklists for the audit tasks

Total Laboratory Hours	**30 hours**

Recommended by Board of Studies	05.02.2020		
Approved by Academic Council	58	Date	26.02.2020

MAT3001	Advanced Mathematics	L	T	P	J	C
		3	1	0	0	4
Pre-requisite	MAT1011 - Calculus for Engineers	Syllabus Version				
		1.0				

Course Objectives:

1. The objective of this course is to give a presentation of basic concepts of linear algebra to illustrate its power and utility through applications to computer science and engineering. Transform techniques are useful in the analysis of signals in communication engineering.
2. By the end of the course the students are expected to learn the concepts of vector space, linear transformations, matrices and inner product space. Further the students are expected to solve problems in cryptography, computer graphics and Fourier and wavelet transforms.

Expected Course Outcome:

1. Observe the various method to solve the system of linear equations and eigen value problems solved by iterative methods
2. Understand the concepts of Vector spaces, Basis and finite dimensional vector spaces

Module:1	System of Linear Equations	5 hours

Gauss-Jacobi, Gauss-Seidel iterative methods for solutions of linear systems and their rates of convergence. Generalized conjugate gradient, Krylov space and Lanczos methods.

Module:2	Iterative methods	6 hours

Symmetric, non-symmetric and generalized eigenvalue problems. Singular value decompositions.

Module:3	Vector Spaces	6 hours

The Euclidean Space – Vector Space – Subspace - linear combination-span-linearly dependent-independent- bases - dimensions-finite dimensional vector space.

Module:4	Linear Transformations	6 hours

Linear transformations – Basic properties - invertible linear transformation- matrices of linear transformations.

Module:5	Vector spaces of Linear Transformations and Applications.	6 hours

Vector space of linear transformation – change of bases – similarity – application to computer graphics.

Module:6	Fourier Transforms	7 hours

Fourier analysis – Fourier and inverse Fourier transforms, uncertainty relation – power spectral density, errors and noise, Discrete Fourier transform – Fast Fourier transform.

Module:7	**Wavelet transform**	**7 hours**
Inversion formula, scaling functions – Haar wavelets – Orthonormal wavelets – wavelet decomposition.		
Module:8	**Contemporary issues**	**2 hours**
Industry Expert Lecture		
	Total Lecture hours:	**45 hours**
Tutorial	• A minimum of 10 problems to be worked out by students inevery Tutorial Class • Another 5 problems per Tutorial Class to be given as home work.	**30 hours**

Text Book(s)

1. C. F. Gerald and P. O. Wheately, –Applied Numerical Analysis‖, 7th ed., Addison Wesley publication, 2015
2. Jin Ho Kwak and Sungpyo Hong, Linear Algebra, Second edition, Springer(2004).(Topics in the Chapters 1,3,4 &5)
3. C. K. Chui, –An Introduction to wavelets‖, Academic press.
4. K. Ogata, –System Dynamics‖, 4th edition., International student edition.

Reference Books

1. L.A. Pipes and L.R. Harvill, "Applied mathematics for Engineers and Physisits‖, TMH Intenational, 3rd Edition.
2. F. B. Hildebrand, –Method of Applied Mathematics‖, 2nd ed., Dover publications.
3. Introductory Linear Algebra- An applied first course, 9th Edition Bernard Kolman and David R. Hill, Pearson Education, 2011.
4. G.H. Golub and C.F. Van Loan, Matrix Computations, North Oxford Academic, 1983.
5. P. Hagedorn, –Nonlinear Oscillations‖, Clarendon Press.
6. Agostino Abbate, C.M.Decusatis, P.K.Das. "Wavelets and Sub-bands- Fundamentals and applications.",Birkhanser (2002).

Recommended by Board of Studies	16.08.2017		
Approved by Academic Council	No. 47th	Date	05.10.2017

MAT3002	Graph Theory and Its Applications	L	T	P	J	C
		3	1	0	0	4
Pre-requisite	MAT2002 Applications of Differential and Difference Equations	Syllabus Version				
		1.0				

Course Objectives:

1. To provide fundamental ideas on graph theory required for the innovate and design applications of Computer Science.

Expected Course Outcome

1. Know the construction of graph model and basic properties of graphs, trees, connectivity and fundamental circuits**.**
2. Understand the planar and dual graphs.
3. Learn about the concepts of matrix representation, matching, coloring and covering on graphs,
4. Analyze the various properties of digraphs and its applications. Construct the graph algorithms for networks and other realistic problems.
5. Construct the graph algorithms for networks and other realistic problems.

Module:1	Graphs and Trees	7 hours

Definition of graphs -subgraphs- Isomorphism - Operations on Graphs - Paths and Cycles - Connected Graphs – Euler and Hamiltonian Graphs -Trees - Some Properties of Trees – Distance and Centre in a tree- Spanning Tree – Rooted and Binary trees.

Module:2	Connectivity and Fundamental Circuits	6 hours

Cut Sets and Cut Vertices - Edge Connectivity and Vertex Connectivity - Fundamental Circuits and Fundamental Cut Sets-Fundamental Circuits.

Module:3	Planar and dual graphs	6 hours

Planar graph - Combinatorial representation**,** Kuratowski‘s graphs, detection of planarity – Dual of a planar graph

Module:4	Matrix Representation and Graph Matching	6 hours

Matrix of a Graph- Incidence Matrix-Adjacency Matrix -Circuit Matrix/Cycle Matrix- Bipartite graphs – Matching –Hall‘s marriage theorem

Module:5	Graph coloring , covering and Partitions	6 hours

Graph coloring- Chromatic number – Chromatic polynomial - Four color Theorem **–** Coverings – Vertex and Edge covering-Partitions.

Module:6	**Digraphs**	**6 hours**
Digraphs – Types of digraphs – Directed paths and connectedness – Euler graphs – Adjacency matrix of a digraph – Tournament		
Module:7	**Graph Algorithms**	**6 hours**
Weighted graph- Shortest path – Shortest path algorithms -Minimum Spanning Tree algorithms-Network flow problem – Max-flow-Min-cut theorem.		
Module:8	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**45 hours**
Tutorial	• A minimum of 10 problems to be worked out by students in every Tutorial Class. • Another 5 problems per Tutorial Class to be given as home work.	**30 hours**

Text Book(s)

1. Santanu Saha Ray, Graph Theory with algorithms and its applications in Applied Science and Technology Springer , 2013.
2. Narsing Deo, Graph Theory with application to Engineering and Computer Science, Prentice Hall India, 2014.

Reference Books

1. D. B. West, Introduction to Graph Theory, 3rd Edition, Prentice-Hall, Englewood Cliffs, NJ , 2007.
2. R. Balakrishnan and K. Renganathan, A Text Book of Graph Theory, Springer, 2012.
3. C. Vasudev, Graph Theory with Application, New Age International (P) Limited, 2006.

Mode of Evaluation

Digital Assignments(Solutions by using soft skill),Quiz, Continuous Assessments, Final Assessment Test

Recommended by Board of Studies	16. 08. 2017		
Approved by Academic Council	No. 47th	Date	05. 10. 2017

SWE1002	Optimization Techniques	L	T	P	J	C
		3	1	0	0	4
Pre-requisite	**None**	**Syllabus version**				
		v. 1.20				

Course Objectives:

1. To understand the role of optimization techniques and its importance in engineering
2. To introduce the concept of linear and nonlinear optimization methods.
3. To realize the application of non-traditional optimization algorithms
4. To choose appropriate optimization method and solve real world problems.

Expected Course Outcome:

1. Comprehend the need and applications of the optimization methods
2. Understand the concept of one-dimensional nonlinear optimization methods.
3. Recognize the unconstrained nonlinear optimization methods.
4. Understand and solve the constrained nonlinear optimization methods.
5. Analyze the concept of quadratic programming and its applications.
6. Apply geometric programming..
7. Comprehend the evolutionary computation techniques for nonlinear programming..

Module:1	**Classical Optimization Techniques**	**6 Hours**

Introduction, methods, engineering applications of optimization-Statement of an optimization problem-classification of optimization problems-Single variable optimization-Multivariable optimization with no constraints-Multi variable optimization with equality and in equality constraints: Lagrange multipliers method, Kuhn-Tucker conditions.

Module:2	**One-Dimensional Nonlinear Optimization**	**6 Hours**

Unimodal function – Region elimination methods: Unrestricted search, Dichotomous Search, Fibonacci method, Golden Section method.

Module:3	**Unconstrained Nonlinear Optimization**	**6 Hours**

Direct Search methods: Univariate method, Pattern directions, Hook and Jeeves‘ method, Powell‘s method-Indirect search methods: Gradient of a function, Cauchy method, Fletcher-Reeves method.

Module:4	**Constrained Non-linear Optimization**	**6 Hours**

Characteristics of a constrained optimization problem - Direct methods: Cutting plane method, methods of feasible directions – Indirect methods: Interior and exterior penalty function methods.

Module:5	**Quadratic programming**	**6 Hours**

Introduction-applications-necessary conditions-solution to quadratic programming problem using Wolfe‘s method.

Module:6	Geometric programming	6 Hours
Introduction to Geometric programming – Solution from differential calculus point of view – Solution from arithmetic-geometric inequality point of view.		
Module:7	**Advanced Non-linear Optimization**	**7 Hours**
Genetic Algorithms -Working principle-Genetic operators-Numerical problem-Simulated Annealing – Numerical problem - Neural network based optimization-Optimization of fuzzy systems-fuzzy set theory-computational procedure		
Module:8	**Contemporary issues**.	**2 Hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1. Singiresu S. Rao, S. S. Rao, Engineering Optimization: Theory and Practice, 2009.

Reference Books

1. C. B Gupta ,Optimization Techniques in Operation Research, I.K.International House Pvt.Ltd 2007.
2. Godfrey C. Onwubolu, B. V. Babu,New Optimization Techniques in Engineering, 2004
3. Cesar Lopez,MATLAB Optimization Techniques,2014
4. **Sherali,** H.D., **Shetty**, C.M.,Optimization with Disjunctive Constraints,Springer,2016(e-book)

Recommended by Board of Studies		12-8-2017	
Approved by Academic Council	No. 47th	Date	5-10-2017

SWE1008	Web Technologies	L	T	P	J	C
		3	0	2	0	4
Pre-requisite	CSE1002	Syllabus version				
		v.1.0				

Course Objectives:

1. To understand the basic technologies, functionality, and applications influencing Web Programming
2. To learn the fundamentals for the web system and internet programming
3. To design and publish web applications using open source software

Expected Course Outcome:

1. Understand the basic structure of the Internet and web page.
2. Learn the fundamentals of *JavaScript* in Web development.
3. Design and develop web pages using CSS styles.
4. Illustrate the basic concepts of PHP in web application.
5. Design and execute dynamic, database-driven web pages using PHP.
6. Understand and apply advanced PHP concepts.
7. Understand the CGI program concepts in PERL.
8. Apply industry-standard tools and frameworks for developing responsive web design.

Module:1	Introduction to HTML5	6 hours

Introduction, Evolution of Web, W3C, HTML5, Headings, Links, Images, Lists, Tables, Frames, Divisions, Forms, Media Tags

Module:2	Java Script	6 hours

Introduction to JavaScript, Variables, Conditional and Loops, Events, Functions, Frames, HTML document, Predefined Object, Image Object, Layers, Drag and Drop

Module:3	Dynamic HTML	6 hours

Introduction to Cascading Style Sheets, Inline Styles, Style Sheets, Grouping & Short Hand Properties, Inheritances, Classes, Link, Cascading Styles, Dynamic Style. Document Object Model.

Module:4	Introduction to PHP	6 hours

History, Basic syntax, Defining functions, Useful functions and language constructs, Arrays, Web, Exceptions, Date and time, Regular expressions

Module:5	MYSQL Database	6 hours

Introduction to MySQL, Data types, Advanced SQL query building, Advanced MySQL Joins, PHP with MySQL, PHP MyAdmin, Importing and Exporting CSV Files

Module:6	Advanced PHP Concepts	6 hours

File Functions, File uploading, Graphics, Mail, Multipart mailing, Attachments in mail, Sessions, Cookies

Module:7	CGI with PEARL	7 hours
Introduction to PERL, Basic I/O, Variables, and Scalar Data, Arrays, Lists, and Hashes, CGI Programming, Pattern Matching.		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1.	Harvey M. Deitel and Paul J. Deitel , –Internet and World Wide Web – How to Program 5th editon‖, Pearson Education, November, 2011.

Reference Books

1. Paul S. Wang, Chapman & Hall "Welcome to Dynamic Web Programming and HTML5"1st Edition CRC Press, Florida, USA, November 21, 2012 ISBN 978-1-4398-7182-9
2. Tom Christiansen, brian d foy, Larry Wall, Jon Orwant "Programming Perl", 4th Edition, O'Reilly Media, February 2012.
3. Kevin Tatroe,Peter MacIntyre,Rasmus Lerdorf –Programming PHP‖ 3rd Edition, O'Reilly Media, July 2014

List of Challenging Experiments (Indicative)

1.	HTML	
2.	DHTML	
3.	java Script	
4.	Form Validations in PHP	
5.	File Handling in PHP	
6	Databases in PHP	
7	Session Tracking in PHP	
8	PERL	
Total Laboratory Hours		45 hours

Recommended by Board of Studies	5-3-2016		
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE1009	.NET Programming	L	T	P	J	C
		3	0	2	0	4
Pre-requisite	CSE1002	Syllabus version				
		v. 1.0				

Course Objectives:

1. To understand the fundamentals of developing modular application using object oriented concepts.
2. To utilize the .NET framework to build distributed enterprise applications
3. To develop console application, windows application and ASP.NET application services.

Expected Course Outcome:

1. Understand the .NET framework to build distributed enterprise application
2. To understand the fundamentals of developing modular application by using objects oriented concepts
3. Comprehend the steps to design, Console Application programs and evaluation of Reflection and attribute based programming
4. Apply an interactive design process and Graphic programming using GDI techniques.
5. Design application for connecting Remote systems using marshaling concepts and socket programming like TCP-UDP using C#
6. Design Data Access with ADO.NET applications by connecting front end and back end through various Data sets
7. Design Web development and ASP.NET application, usage of various web form controls and validation controls.
8. Apply .Net Programming in industries

Module:1	.NET Framework	6 Hours

Common language Runtime (CLR) – Common Type System (CTS) – Common language Specification (CLS) – Compilation process – Assemblies – Namespaces – Command line compiler

Module:2	C# language fundamentals	6 Hours

Programming constructs – value types and reference types – object oriented concepts – Encapsulation – Inheritance – polymorphism – Interfaces – collections – Multithreading

Module:3	Console Application	6 Hours

Indexers - Multicast delegates – Events - Registry programming – File I/O - Serialization – Binary format – SOAP format – Type Reflection and attribute-based programming – Late binding

Module:4	Windows Forms	6 Hours

Tool box controls – Container control – Menu – Tool bar – Tool tip Controls during design time – Run time – Graphics programming GDI+

Module:5	Remoting	6 Hours

Architecture - Marshal By value (MBV) – Marshal By Reference (MBR) – Network programming using C# - Socket – TCP – UDP

Module:6	**Data Access with ADO.NET**	**6 Hours**
Architecture – Data reader – Data Adapter – Command – Connection – Data set – Data binding – Data Grid Control – XML based Data sets		
Module:7	**Web Development and ASP.NET**	**7 Hours**
Architecture – web forms – web form controls – Life time Management - Application – Session – ASP with ADO.NET Validation controls – website security		
Module:8	**Contemporary issues**	**2 Hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1. Pro C# 5.0 and the .NET 4.5 Framework , 6th edition, Andrew Troelsen, APress., 2012

Reference Books

1. C# in depth, Joh Skeet, Manning publications , 3rd edition , 2014
2. Head First C#, Adrew Stellman and Jennifer Greene, 3rd edition, O'Reilly, 2013

List of Challenging Experiments (Indicative)

1.	Write a program using c# to create a DLL for laptop object with necessary types such as methods, fields, property etc.Create a windows form to display the various types available in laptop object using the concept of Reflection. [Hint: Store the count of types in registry]	
2.	Create a DLL for ATM Object with necessary fields, properties and methods such as initiating, deposit and withdrawal. Write a menu driven program to perform the following, (i) Discover all the types that are available in the DLL using the concept of multicast delegates. (ii) After initiating the basic information of the customer perform serialization using SOAP format. (iii) Deserialize the above and invoke the methods such as deposit and withdrawal using the concept of late binding. While performing withdrawal, check for the minimum balance value that has to be retrieved from registry.	
3.	Create a DLL Sum with overloaded methods such as, Sum_a(double s, double t); Sum_a(int i, int j); Sum_a(int k, double b); Write a menu driven program to perform the following, (i) Discover all the types that are available in the DLL using the concept of	

	multicast delegates. (ii) After initiating the values perform serialization using Binary format. (iii) Deserialize the above and invoke the methods using the concept of late binding. If the signature of a method which is invoked is (double, double) then store the result value in registry.	
4.	Create a DDL for Calculator with basic operation such as add, sub, multiply and divide. All the methods defined in the calculator should have a return type. Using the concept of multicast delegates & get invocation list () invoke the methods in calculator object.	
	Total Laboratory Hours	30 hours

Recommended by Board of Studies		5-3-2016	
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE1010	Digital Image Processing	L	T	P	J	C
		3	0	0	4	4
Pre-requisite	MAT1011	Syllabus version				
		v. 1.0				

Course Objectives:

1. Introduce the concept of digital image and the fundamental steps in digital image processing
2. Learn applying basic image processing techniques for developing specific image processing systems.
3. Comprehend the steps of experimental design for a particular problem domain and demonstrate the system of image processing.

Expected Course Outcome:

1. Understand the concepts of image acquisition and digitization .
2. Classify image enhancement techniques and apply these techniques in both spatial and frequency domain.
3. Recognize the types of noise present in images and apply appropriate image restoration technique.
4. Categorize image segmentation techniques and apply these techniques
5. Study the importance of image compression and apply basic compression techniques to images.
6. Analyse various image representation techniques & descriptors and understand its importance to computer vision.
7. Implement basic morphological image processing techniques on images and understand color models for images
8. Learn digital image processing stepts and apply appropriate techniques to a specific problem domain.

Module:1	DIGITAL IMAGE FUNDAMENTALS	6 hours

Introduction, Digital Image Fundamentals, image acquisition and display using digital devices - Human visual perception, properties -Image sampling and quantization-Basic relationship between pixels.

Module:2	IMAGE ENHANCEMENT	8 hours

Image enhancement in the spatial domain: basic grey level transformation, Histogram Processing-Enhancement using arithmetic/Logic operations-Spatial filtering: smoothing and sharpening. Image enhancement in the frequency domain: Introduction to two-dimensional transforms-Discrete Fourier Transform, Discrete Cosine Transform, Discrete Wavelet Transform – smoothing frequency domain filtering-sharpening frequency domain filtering.

Module:3	**IMAGE RESTORATION**	**5 hours**
Noise Models-Restoration in the presence of Noise only-spatial filtering-periodic noise reduction by frequency domain filtering.		
Module:4	**IMAGE SEGMENTATION**	**8 hours**
Detection of discontinuities, Edge Linking and Boundary Detection, Thresholding Methods, Region Oriented Methods.		
Module:5	**IMAGE COMPRESSION**	**5 hours**
Lossless Image Compression- The Concept of entropy and Huffman coding; Run-length coding for grey images,Lossy Image Compression – Predictive coding, transform coding – JPEG compression standard, Wavelet-based image compression JPEG2000.		
Module:6	**REPRESENTATION AND DESCRIPTION:**	**5 hours**
Chain codes, Polygonal approximation, Signature Boundary Segments, Skeltons, Boundary Descriptors, Regional Descriptors, Relational Descriptors, Principal components for Description, Relational Descriptors.		
Module:7	**MORPHOLOGICAL AND COLOR IMAGE PROCESSING**	**6 hours**
Dilation and Erosion-Opening and Closing-Hit or Miss Transformation-Basic morphological algorithms.Color Image processing: Light and color, color formation, Colour models, Histogram of a color Image, Color image filtering, Gamma correction and segmentation of color image.		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1. R.C. Gonzalez & R.E. Woods,―Digital Image Processing‖ , Pearson Education, Third Edition,2013

Reference Books

1. S. Jayaraman, S. Esakirajan & T.Veerakumar ― Digital Image Processing‖, Tata Mcgraw-Hill First Edition 2009.
2. A. K. Jain**,** ―Fundamentals of Digital Image Processing," Pearson Education (Asia) Pte. Ltd./Prentice Hall of India, 2004.
3. Jhon C Ross, ― The Image Processing Hand Book‖, CRC Press 5th Edition,2006
4. B. Chanda and D. Dutta Majumdar ―Digital Image Processing and Analysis‖, PHI, 2011.

Recommended by Board of Studies		5-3-2016	
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE1011	Soft Computing	L	T	P	J	C
		3	0	0	4	4
Pre-requisite	**MAT1013**	**Syllabus version**				
		v. 1.0				

Course Objectives:

1. To understand the fundamentals of neural network and its applications
2. To learn about the concept of fuzzy logic components
3. To expose the ideas about genetic algorithm

Expected Course Outcome:

1. Understand the basics of artificial neural network and supervised learning network
2. Apply knowledge and understanding of associative memory networks
3. Apply knowledge and understanding of unsupervised learning network
4. Comprehend fuzzy sets and of fuzziness involved in various systems
5. Understand the concepts of fuzzy logic, knowledge representation
6. Understand fuzzy concepts and develop a Fuzzy inference system to derive decisions.
7. Understand the concepts of genetic Algorithm
8. Apply soft computing techniques for real life applications

Module:1	**Neural networks**	**7 hours**

Introduction to Soft computing, basics. Neural networks, introduction, evolution, basic models, terminologies of ANN, Pitts model, Perceptron, Adaline, Back-propagation network, RBF network.

Module:2	**Memory Models**	**5 hours**

Pattern association, auto & hetero associative memory models, Radial Basis Function, BAM, Hopfield network

Module:3	**Unsupervised Networks**	**6 hours**

Kohonen Self-organizing maps, LVQ network, ART, Recurrent networks and deep learning

Module:4	**Fuzzy sets**	**6 hours**

Introduction, fuzzy sets and crisp sets, operations, fuzzy relations, fuzzification & defuzzification

Module:5	**Fuzzy logic and approximate reasoning**	**7 hours**

Membership functions, Fuzzy truth values, fuzzy propositions, fuzzy rules, formation, decomposition and aggregation of rules, fuzzy reasoning.

Module:6	**Fuzzy Decision making**	**6 hours**
FIS, Fuzzy controller. Individual decision making, multi-objective and multi-attribute decision making, Industrial applications.		
Module:7	**Search Strategies**	**6 hours**
Basic concepts of search strategies, Genetic Algorithm working principle, procedures of GA, flow chart of GA, Genetic representations, (encoding) Initialization and selection, Genetic operators, Mutation, Generational Cycle, Applications		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1. Principles of Soft Computing, 2nd Edition by Sivanandam & Deepa, Wiley India, 2011.

Reference Books

1. Introduction to Soft Computing, by Samir Roy and Udit Chakraborty, Pearson, 2013
2. Fundamentals of Neural networks: architectures, algorithms and applications by Laurene Fausett, Pearson India, 2008
3. Fuzzy logic with Engineering Applications, 3rd Edition by T.J. Ross, Wiley India, 2010

	Recommended by Board of Studies		5-3-2016	
	Approved by Academic Council	No. 40th	Date	18-3-2016

SWE1012	E-Governance	L	T	P	J	C
		2	0	0	4	3
Pre-requisite	**None**	**Syllabus version**				
		v. 1.0				

Course Objectives:

1. To gain critical understanding of e-governance with multidisciplinary view.
2. To learn how to use ICT in public governance systems.
3. To understand the design and evaluation various E Governance frameworks

Expected Course Outcome:

1. Analyze the basics of e-governance in particular National e-governance plan.
2. Apply the concepts of e-governance in various applications.
3. Recognize the concepts of process reengineering and change management.
4. Select and Apply the various technologies in e Governance projects.
5. To create or setup the required infrastructure for e governance projects
6. Identify and choose the open standards for e-governance.
7. Use various tools used for e governance
8. Design and develop citizen centric systems

Module:1	Overview of e-Governance	5 hours

National and International Governance, e-Government and e-Governance, India's National e-Governance Plan (NeGP), Preparing for e-Governance, Stakeholders consultation and service identification

Module:2	e-Governance project life cycle	5 hours

E-Governance applications in selected Government sectors, -Health, Local Body Administration, Education, Agriculture, Land Records, etc., Process Re-engineering- Process Reengineering and change management, e-Governance system design. e-Governance project life cycle and project management

Module:3	Technologies for e-Governance	6 hours

Data warehousing, data mining, geographical information systems, biometrics, smartcards, cloud computing and virtualization, web portals.

Module:4	e-Governance eco system	6 hours

E-Governance ecosystem, e-Governance infrastructure–Data Centers, National Optical Fiber Network (NOFN)

Module:5	E Governance Networks	6 hours

State-Wide Area Networks (SWANs), National/State/District portals, Open Standards based e-Governance. Case studies on e-Governance-Monitoring and evaluation tools for e-Governance, Assessing learning outcomes of e-Governance projects.

Module:6	**Contemporary issues**			**2 hours**
	Total Lecture hours:			**30 hours**
Text Book(s)				
1.	C.S.R. Prabhu, E-Governance: Concepts and Case Studies, Prentice-Hall of India, Second Edition, 2013.			
Reference Books				
1.	D.N. Gupta, E-Governance: A Comprehensive Framework, New Century Publications, First Edition 2008.			
2.	Abdelbaset Rabaiah, Best-Practice Framework for Developing and Implementing E-Government, VUB Press, Second Edition, 2009.			
	Recommended by Board of Studies		5-3-2016	
	Approved by Academic Council	No. 40th	Date	18-3-2016

SWE1013	Multimedia Systems	L	T	P	J	C
		2	0	0	4	3
Pre-requisite	**None**	**Syllabus version**				
		v. 1.20				

Course Objectives:

1. To gain the knowledge in broadcasting, audio recording, media, mass communication and digital animation
2. To Equip students in art and craft of multimedia production as to enable them to emerge as thoroughbred professionals matching the needs of fast growing multimedia industry
3. To develop and analyze the performance of multimedia communication systems

Expected Course Outcome:

1. Analyze the technical aspects of Graphics and Multimedia systems.
2. Understand data interface standards for text, image, graphics, audio, video and animation
3. Apply image representation and compression concepts in real world Multimedia applications.
4. Design interactive multimedia software using audio representation and compression concepts
5. Apply various multimedia communication protocols and standards.
6. Evaluate multimedia application for its optimum performance
7. Use multimedia authoring tools for industry requirements
8. Design multimedia system for the productive use of social media

Module:1	Introduction to Computer Graphics	3hours

Introduction to Computer Graphics, Two dimensional concepts and Transformations, Three dimensional concepts and Transformations

Module:2	Multimedia Communication and Standards	5 hours

Concept of multimedia communication modeling – elements for multimedia systems – network requirements – text, audio, images and video – multimedia processing in communication – distributed multimedia systems, MPEG -1, 2, 4, JPEG -2000, MPEG-7,21 and Internet standards.

Module:3	Image Representation and Compression	8 hours

Color model in images-lossless compression algorithms- run-length encoding, variable length coding, dictionary based coding, arithmetic coding, lossy compression algorithms- quantization, transform coding, wavelet-based coding- **Multimedia Authoring Tools-** Overview of multimedia software tools, Multimedia Authoring systems, editing and authoring tools, hypermedia application design considerations, VRML

Module:4	Audio Representation and Compression	4 hours

Digitization of sound, MIDI, transmission of audio, audio compression techniques- ADPCM, vocoders

Module:5	Video Representation Compression	8 hours
Color model in video, types of video signals, analog and digital video, video compression techniques- based on motion compensation, intra-frame coding, inter-frame predictive coding, H.263- **Multimedia Network Communication and Applications-** Quality of Multimedia data transmission, Multimedia over IP, Multimedia over ATM networks, media-on-demand		
Module:6	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**30 hours**

Text Book(s)

1. Multimedia Communication Systems, Techniques, Standards and networks‖, Kamisetty Ramamohan Rao, Z.S.Bojkovic,D.A.Milovanovic,PHI learning, 2012.

Reference Books

1. Multimedia Applications‖, Ralf Steinmetz and klara Nahrstedt, 2004
2. Multimedia and Applications‖, Hemant Kapila, 2016
3. Multimedia systems design‖, Prabhat k. Andleigh, Kiiran Thakrar, PHI learning, 2010
4. Fundamentals of multimedia‖ Ze-Nian, Mark S. Drew, PHI learning, 2010
5. Multimedia: Making it Work‖, Tay Vaughan, Eighth edition, 2011

Recommended by Board of Studies	12-8-2017		
Approved by Academic Council	No. 47th	Date	5-10-2017

SWE1014	Enterprise Resource Planning	L	T	P	J	C
		2	0	0	4	3
Pre-requisite	**None**	**Syllabus version**				
		v. 1.0				

Course Objectives:

1. To understand the fundamental concepts of ERP systems, their architecture and working of different modules in ERP
2. To prepare the students technological competitive and make them ready to self-upgrade with the higher technical skills
3. Focus on a strong emphasis upon practice of theory in applications and practical-oriented approach

Expected Course Outcome:

1. Understand the functional Areas and business Processes of ERP
2. Comprehend the significance and benefits of ERP Software
3. Study the Marketing and Information Systems and the Sales Order Process
4. Study the production and Supply Chain Management Information Systems
5. Design accounting module for a given case study.
6. Indentify the features of Human Resource Process
7. Use tools and techniques required for implementation of ERP.
8. Enumerate the applications of ERP in different sectors

Module:1	Business Functions-Business Processes	5 hours

Functional Areas and Business Processes-Functional Areas and Business Processes of Very Small Business-Functional Area Information Systems, ERP Systems - The Evolution of Information Systems-ERP Software Emerges: SAP and R/3- ERP for Midsized and Smaller Companies

Module:2	Marketing Information Systems and the Sales Order Process	5 hours

Fitter Snacker-Problems with Fitter Snacker‘s Sales Process-Sales and Distribution in ERP-A Standard Order in SAP ERP-Customer Relationship Management (CRM). Production Overview-The Production Planning Process-ERP and Suppliers.

Module:3	Production and Supply Chain Management Information Systems	6 hours

Production module- Fitter‘s Manufacturing Process- Fitter‘s Production Problems- The Production Planning Process- The SAP ERP Approach to Production Planning- Sales Forecasting- Demand Management- Materials Requirements Planning (MRP)- ERP and Suppliers

Module:4	Accounting in ERP Systems	6 hours

Accounting Activities- Operational Decision-Making Problem: Credit Management- Product Profitability Analysis- Management Reporting with ERP Systems- Trends in FinancialReporting—XBRL. Case Study: ENRON

Module:5	**Human Resource Process**	**6 hours**
Problems with Fitter‘s Human Resources Processes-Human Resources with ERP Software-Advanced SAP ERP Human Resources Features-Additional Human Resources Features of SAP ERP, ERP Implementation		
Module:6	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**30 hours**

Text Book(s)	
1.	Ellen F. Monk, Bret J. Wagner, Concepts In Enterprise Resource Planning, 4th Edition, Cengage Learning, 2013.

Reference Books

1. Alexis Leon ,ERP Demystified, Third Edition , Tata McGraw Hill, 2014.
2. Ganesh, K., Mohapatra, S., Anbuudayasankar, S.P.,Sivakumar, P., Enterprise Resource Planning, Fundamentals of Design and Implementation, Springer, 2014.

Recommended by Board of Studies	5-3-2016		
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE1015	Biometric Systems	L	T	P	J	C
		2	0	0	4	3
Pre-requisite	MAT2001	Syllabus version				
		v. 1.0				

Course Objectives:

1. To understand design process of large scale biometric identification Systems.
2. To analyze problems in various biometric traits.
3. To design biometric systems from sensor to decision.
4. To Construct and evaluate the multimodal biometric Systems.

Expected Course Outcome:

1. Comprehard the concepts and terminology of biometric recognition system
2. Distinguish among various Biometric Technologies along with their advantages and disadvantages
3. Develop various biometric modality authentication systems
4. Improve existing algorithms used in personal authentication systems
5. Analyse Multi biometrics systems and applications
6. Identify and choose different evaluation techniques for biometric systems
7. Design of effective and secure biometric authentication system
8. Illustrate the applications of biometric systems in industry

Module:1	Introduction of Biometrics	5 hours

Introduction, Fundamental of Technical Evaluations, Types of errors, Performance Metrics, Evaluation Methodologies, Design of Evaluation.

Module:2	Fingerprint Recognition	5 hours

Fingerprint Anatomy, History, Fingerprint Presentation and acquisition, Fingerprint Feature Extraction, Fingerprint Feature Matching, Automated Fingerprint Identification System.

Module:3	Face Recognition and Iris Recognition	6 hours

History, 2D Face Recognition -Face Presentation and acquisition, Feature Extraction and Matching, 3D Face Recognition, Iris Anatomy, History, Iris image acquisition, Iris Feature Extraction, Iris Feature Matching.

Module:4	Behavioral Biometrics and Multi Biometrics	6 hours

Hand geometry, Palm print, Dynamic Signature, Keystroke, Ear, DNA Voice and Gait,Need for Multi biometrics, Multi biometric system design, Data acquisition, Levels of fusion.

Module:5	Biometric Testing and Security	6 hours

Needs of Biometric testing, Biometric data considerations, Unimodal Performance Evaluation and Multimodal Performance Evaluation, Comparative tests, Biometric system security.

Module:6	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**30 hours**

Text Book(s)

1. Shimon K. Modi, Biometrics in Identity Management: Concepts to Applications, Artech House, 2011

Reference Books

1. G.R. Sinha, Sandeep B. Patil, Biometrics: Concepts and Applications, Wiley, 2013.
2. James L. Wayman, Anil Jain, DavideMaltoni, Dario Maio, Biometric Systems: Technology, Design and Performance Evaluation, Springer 2010.
3. Anil Jain, Patrick Flynn, Arun Ross, Handbook of Biometrics, Springer, 2008.

Recommended by Board of Studies	5-3-2016		
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE1017	Natural Language Processing	L	T	P	J	C
		2	**0**	0	4	**3**
Pre-requisite	**SWE1006**	**Syllabus version**				
		v 1.20				

Course Objectives:

1. To understand principles processing
2. To apply phonological, morphological and syntactic processing techniques to process linguistic data.
3. To develop mathematical models for information retrieval.

Expected Course Outcome:

1. Understand preprocessing techniques to prepare the text data for text processing and information extraction applications.
2. Understand methods and algorithms used to process different types of textual data as well as the challenges involved.
3. Build generic computational models for word-form recognition and Production
4. Design a parser for text to structured representation mapping
5. Develop an application to interlink words in text by means of conceptual-semantic and lexical using WordNet lexical database.
6. Design and implement a text analysis/retrieval system to visualize the attitude of a user towards a product, topic and etc.
7. Develop computational skills to create NLP processing pipelines using existing NLP libraries, retrain models and extend existing NLP tools
8. Apply evaluation techniques to validate NLP systems

Module:1	**Overview of Natural Language Processing(NLP)**	**5 hours**

Introduction to Natural Language Understanding–NLP Overview: Prerequisite technologies-Subfields of NLP-Related fields of NLP- Structures used in NLP

Module:2	**Sound**	**5 hours**

Biology of Speech Processing-Place and Manner of Articulation-Word Boundary Detection-Argmax based computations-HMM and Speech Recognition

Module:3	**Words and Word Forms**	**6 hours**

Morphology fundamentals-Morphological Diversity of Indian Languages- Morphology Paradigms-Finite State Machine Based Morphology-Automatic Morphology Learning-Shallow Parsing-Named Entities-Maximum Entropy Models

Module:4	**Syntax and Semantics**	**6 hours**

Theories of Parsing-Parsing Algorithms-Robust and Scalable Parsing on Noisy Text as in Web documents-Hybrid of Rule Based and Probabilistic Parsing- Scope Ambiguity and Attachment

Ambiguity resolution- Lexical Knowledge Networks				
Module:5	**Web 2.0 Applications**			**6 hours**
Sentiment Analysis; Text Entailment-Robust and Scalable Machine Translation- Question Answering in Multilingual Setting-Cross Lingual Information Retrieval (CLIR)- Tokenizing Text and WordNet Basics- Replacing and Correcting Words- Part-of Speech Tagging- Extracting Chunks- Text Classification				
Module:6	**Contemporary issues**			**2 hours**
	Total Lecture hours:			**30 hours**
Text Book(s)				
1.	Daniel Jurafsky and James H. Martin –Speech and Language Processing‖, 3rd edition, Prentice Hall, 2013.			
Reference Books				
1.	Allen, J.,‖ Natural Language Understanding‖, 2nd Edition(Reprint), Benjamin/Cummings Publishing Company, 2012			
2.	Chris Manning and Hinrich Schütze, –Foundations of Statistical Natural Language Processing‖, 2nd edition, MIT Press Cambridge, MA, 2015.			
3.	Nitin Indurkhya, Fred J. Damerau –Handbook of Natural Language Processing‖, 2nd Edition, CRC Press, 2010			
4.	Jacob Perkins,‖Python Text Processing with NLTK 2.0 Cookbook‖,1st Edition, PACKT Publishing,2010			
5.	Bing Liu,‖Sentiment Analysis and Opinion Mining, Morgan &Claypool Publishers, May 2012.			
	Recommended by Board of Studies		12-8-2017	
	Approved by Academic Council	No. 47th	Date	5-10-2017

SWE1018	Human Computer Interaction	L	T	P	J	C
		2	0	0	4	3
Pre-requisite	**None**	**Syllabus version**				
		v.1.0				

Course Objectives:

1. To understand guidelines, principles, and theories influencing human computer interaction.
2. To synthesize mock ups and carry out user and expert evaluation of interfaces
3. To comprehend the steps of experimental design, and evaluation of human computer interaction systems.

Expected Course Outcome:

1. Identify the capabilities of both humans and computers from the viewpoint of human information processing.
2. Understand the guidelines and design process for designing HCI systems.
3. Study human–computer interaction (HCI) models, styles, and HCI paradigms.
4. Apply an interactive design process and universal design principles for designing HCI systems.
5. Design a user interface complying with HCI design principles, standards and guidelines.
6. Identify and choose from a variety of user research and evaluation techniques
7. Identify HCI issues in groupware, ubiquitous computing, virtual reality, multimedia, and Word Wide Web-related environments.
8. Apply evaluation and usability testing methods for validating interactive products

Module:1	**Introduction to Human Computer Interaction**	**5 hours**

Human Computer Interaction and its frameworks, Principles of HCI, Types of Interaction styles, HCI Guidelines.

Module:2	**Human factors as HCI Theories**	**6 hours**

Human Information Processing – Task Modeling and Human Problem Solving model; Human Reaction and Prediction of Cognitive Performance; Sensation and Perception of Information; Human Body Ergonomics

Module:3	**HCI Design**	**5 hours**

Interface Selection Options, Wire-Framing, Naïve Design Example.

Module:4	**User Interface Layer and Methodology**	**6 hours**

User interface layer and its execution Framework, Input /Output processes, UI Development Toolkit, Interactive System development Framework, Case studies on MVC.

Module:5	**Evaluation Techniques**	**6 hours**
Goals and types of Evaluation, Evaluation through Expert analysis, Evaluation through user Participation, Choosing an evaluation method.		
Module:6	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**30 hours**

Text Book(s)

1.	Gerard Jounghyun Kim, Human Computer Interaction – Fundamentals and Practice, – CRC press, 2015.

Reference Books

1. Julie A. Jacko, The Human–Computer Interaction Handbook: Fundamentals, Evolving Technologies, and Emerging Applications, 3rd Edition, CRC Press (Taylor & Francis Group) 2012.
2. Ben Shneiderman, Catherine Plaisant, Maxine Cohen, Steven Jacobs, Designing the User Interface: Strategies for Effective Human Computer Interaction, 5th Edition, Pearson, 2009.
3. Alan Dix, Janet E. Finlay, Gregory D. Abowd, Russell Beale, Human - Computer Interaction, 3rd Edition, Pearson, 2003.

Recommended by Board of Studies	5-3-2016		
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE2008	Android Programming	L	T	P	J	C
		3	0	0	4	4
Pre-requisite	**SWE1007**	**Syllabus version**				
		v. 1.0				

Course Objectives:

1. To learn the fundamentals of Android OS Architecture and working principles
2. To understand mobile application development process for Android platform.
3. To comprehend the steps of App design, test, and deployment using Android SDK

Expected Course Outcome:

1. Understand the Android platform, its Architecture and working environment.
2. Learn the Anatomy of an Android app and its core components.
3. Design creative user interfaces for Android app.
4. To learn various storage options in Android to store various types of user data.
5. Apply the software development life cycle to Android app
6. Test an Android app and publish it in the play store
7. Solve real-life problems using android programming
8. Understand industry best practices for mobile app development

Module:1	**Introducing Android**	**6 hours**

Android Development Environment setup, Essentials of Writing Android Application

Module:2	**Android Application Basics**	**6 hours**

Anatomy of an Android Application, Application Using the Android Manifest File, Managing Application Resources

Module:3	**Android User Interface Design Essentials:**	**6 hours**

User Interface Building Blocks, Designing with Layouts, Partitioning the User Interface with Fragments, Displaying Dialogs

Module:4	**Android Application Design Essentials**	**6 hours**

Android Preferences, Files and Directories, Content Providers, Designing Compatible Applications

Module:5	**Software Methodology**	**8 hours**

Mobile Development Process, Choosing Software Methodology, Gathering requirements and assessing risks, Configuration Management, Designing and developing Mobile Applications, Testing and deploying mobile applications, Supporting and maintaining mobile applications

Module:6	**Testing and Publishing**	**5 hours**

Testing Mobile Applications, Android Application Testing Fundamentals, Publishing Android Application

Module:7	**Android Applications**	**6 hours**

Location and Mapping, Drawing 2D and 3D Graphics, Inter Process Communication, Simple Phone Calls.

Module:8	**Contemporary issues**	**2 hours**
.		
	Total Lecture hours:	**45 hours**

Text Book

1.	Joseph Annuzzi, Jr., Lauren Darcey, Shane Conder, ―Introduction to Android Application Development‖, Create Space Independent Publishing Platform, Fourth Edition, 2014.

Reference Books

1. Wei-Meng Lee, Beginning Android 4 Application Development, Wrox, 2012
2. Budi Kurniawan. Introduction to Android Application Development, 2014
3. Dawn Griffiths, Head First Android Development, O'reilly, 2015
4. Rajiv Ramnath, Roger Crawfis, and Paolo Sivilotti, Android SDK 3 for Dummies, Wiley, 2011
5. Rick Rogers, John Lombardo, Zigurd Mednicks and Blake Meike, ―Android Application Development ― , First Edition, 2009.

	Recommended by Board of Studies		5-3-2016	
	Approved by Academic Council	No. 40th	Date	18-3-2016

SWE2009	Data Mining Techniques	L	T	P	J	C
		3	0	**0**	4	**4**
Pre-requisite	**SWE1004**	**Syllabus version**				
		v 1.0				

Course Objectives:

1. To understand the fundamental data mining methodologies and with the ability to formulate and solve problems.
2. To classify data mining systems and understand methods for data gathering and data pre - processing.
3. To learn data mining techniques, for solving real world problems

Expected Course Outcome:

1. Understand the basics of data techniques and their applications real world scenarios.
2. Apply frequent pattern analysis in business analytics
3. Design appropriate classification techniques and association rule generation.
4. Comprehend clustering techniques and discover the knowledge imbibed in the high dimensional system.
5. Deploy of advanced classification techniques in real world applications.
6. Comprehend and use the specific clustering approaches
7. Develop applications targeted for real world problems based on advanced data mining techniques.
8. Design and develop an information retrieval system using various data mining approaches for a given problem.

Module:1	**Data Mining Concepts :**	**6 hours**

Introduction to Data Mining – Data Mining Functionalities – Classification of Data Mining Systems, Data Mining Task Primitives-Integration of Data Mining With Database- Major Issues in Data Mining.

Module:2	**Frequent Pattern Mining:**	**6 hours**

Basic Concepts – Market Basket Analysis - Efficient and Scalable Frequent Item Set Mining Methods – The Apriori Algorithm – Frequent Pattern Growth Algorithm-Various Kinds of Association Rules- Association Mining to Correlation Analysis.

Module:3	**Classification and Prediction:**	**6 hours**

Classification - Issues Regarding Classification and Prediction -Decision Tree Induction- Bayesian Classification - Rule-Based - Accuracy and Error Measures.

Module:4	**Cluster Analysis:**	**6 hours**

Types of Data in Cluster Analysis - Major Clustering Methods- The K-Means Method.

Module:5	**Clustering:**	**6 hours**
Similarity and Distance Measures- Hierarchical Algorithms- Partitioning Algorithms- Clustering Large Databases- Clustering with Categorical Attributes.		
Module:6	**Outlier Analysis**	**6 hours**
Outlier Analysis- Distance-Based Outlier Detection- Density-based Local Outlier Detection		
Module:7	**Advanced Techniques**	**7 hours**
Hybrid Techniques- Adaptive Neuro Fuzzy Inference System - Web Mining- Spatial Mining and Temporal Mining.		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1.	J. Han and M. Kamber. Data Mining: Concepts and Techniques- 3rd Edition. Morgan Kaufman. 2011.

Reference Books

1.	**Pang-Ning Tan , Michael Steinbach and Vipin Kumar. Introduction to Data Mining, Pearson, 2014.**
2.	M. H. Dunham. Data Mining: Introductory and Advanced Topics. Pearson Education. 2001.

Recommended by Board of Studies		5-3-2016	
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE2010	Embedded Systems	L	T	P	J	C
		2	0	0	4	3
Pre-requisite	SWE1003	Syllabus version				
		v. 1.10				

Course Objectives:

1. To discuss the architecture of an embedded system and its components
2. To develop a system for an industry problems on an embedded platform
3. To understand the programming environment for an embedded applications.
4. To learn RTOS concepts, features and classification

Expected Course Outcome:

1. Summarize the key concepts of an embedded systems and its applications.
2. Analyze the communication protocols in an embedded systems with types, advantages and disadvantages.
3. Design and development of hardware, software and firmware for a diversified applications.
4. Apply task scheduling, Multitasking and priority levels in embedded RTOS.
5. Test Inter Task Communication for concurrency in real-time applications.
6. Understand the concepts and basic architecture of microcontroller.
7. Develop Programming skills to create the microcontroller based applications.
8. Interpret the challenges and issues of designing an embedded system applications.

Module:1	Introduction to Embedded Systems	3 hours

History of Embedded Systems, Classification, Major Application Areas, Purpose and Definition of Embedded System, Embedded Systems Vs General Computing

Module:2	Typical Embedded System:	3 hours

Memory: ROM, RAM, Memory according to the type of Interface, Communication Interface: Onboard and External Communication Interfaces.

Module:3	Embedded Firmware:	6 hours

Reset Circuit, Brown-out Protection Circuit, Oscillator Unit, Real Time Clock, Watchdog Timer, Embedded Firmware Design Approaches and Development Languages.

Module:4	RTOS Based Embedded System Design:	6 hours

Operating System Basics, Types of Operating Systems, Tasks, Process and Threads, Multiprocessing and Multitasking, Pre-emptive multitasking, Task Scheduling.

Module:5	**Task Communication:**	**3 hours**
Shared Memory, Message Passing, Remote Procedure Call and Sockets.		
Module:6	**Introducing the 8051 Microcontroller Family**	**3 hours**
Introduction, Clock frequency and performance, Memory issues, I/O pins, Timers, Interrupts, Serial interface.		
Module:7	**Programming Embedded Systems in keil C**	**4 hours**
Introduction to Embedded C, Programming with keil C, Usage with ports and interfaces.		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**30 hours**

Text Book(s)

1.	Dr. K V K K Prasad, ‒Embedded / Real-Time Systems: Concepts, Design And Programming, Black Book‖ , DreamTech Press, 2013.

Reference Books

1. The 8051 Microcontroller And Embedded Systems Using Assembly And C, 2/E. Front Cover. Mazidi. Pearson Education, 2011.
2. Wayner Wolf, ‒Computers as components – Principles of embedded computing system design‖, Morgan Kaufman, 2012.
3. Arnold S Berger, ‒Embedded Systems Design An Introduction to Processes, Tools & Techniques‖, CMP books 2010.

Recommended by Board of Studies		12-8-2017	
Approved by Academic Council	No. 47th	Date	5-10-2017

SWE2011	Big Data Analytics	L	T	P	J	C
		3	0	0	4	4
Pre-requisite	SWE1004	Syllabus version				
		v.1.0				

Course Objectives:

1. To introduce fundamental concepts of big data analytics.
2. To elucidate different data learning techniques.
3. To explore various data analytic and visualization tools.

Expected Course Outcome:

1. Understand characteristics and sources of big data.
2. Recognise of various data analytical techniques and approaches for handling big data.
3. Apply data analytic methodologies in streaming data.
4. Familiar with diverse learning models and clustering techniques.
5. Use visualization techniques and tools in big data analystics
6. Campare the different types of frameworks and tools for big data analytics
7. Analyze Big Data in various forums like Social Networks, e-Commerce etc
8. Illustrate the phases of Big Data Analytics with the help of Data Sets from various domains and presenting the results.

Module:1	Introduction to Big Data	7 hours

Analytics – Nuances of big data – Value – Issues – Case for Big data – Big data options Team challenge – Big data sources – Acquisition – Nuts and Bolts of Big data. Features of Big Data - Security, Compliance, auditing and protection - Evolution of Big data – Best Practices for Big data Analytics - Big data characteristics - Volume, Veracity, Velocity, Variety

Module:2	Data Analysis and Approaches	7 Hours

Evolution of analytic scalability – Convergence – parallel processing systems – analytic data sets – Analytic methods - Analysis approaches – Statistical significance – business approaches – Analytic innovation – Traditional approaches – Iterative

Module:3	Stream Data Mining	5 hours

Introduction to Streams Concepts – Stream data model and architecture - Stream Computing, Sampling data in a stream – Filtering streams – Counting distinct elements in a stream – Estimating moments – Counting oneness in a window – Decaying window – Real time Analytics Platform(RTAP) applications.

Module:4	Predictive Analytics	8 hours

Predictive Analytics – Supervised – Unsupervised learning – Neural networks – Kohonen models – Normal – Deviations from normal patterns – Normal behaviors – Expert options – Variable entry - Mining Frequent itemsets - Market based model – Apriori Algorithm – Handling large data sets in Main memory – Limited Pass algorithm – Counting frequent itemsets in a stream – Clustering Techniques – Hierarchical – K- Means.

Module:5	**Visualizations**	**5 hours**
Clustering high dimensional data Visualizations - Visual data analysis techniques, interaction techniques; Systems and applications.		
Module:6	**Framework for implementation**	**6 hours**
Map Reduce Framework - Hadoop – Hive - – Sharding – NoSQL Databases - S3 -Hadoop Distributed file systems – Hbase – Impala.		
Module:7	**Big Data for E-Commerce**	**5 hours**
Analyzing big data with twitter – Big data for E-commerce – Big data for blogs.		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1. Vignesh Prajapati, Big data analytics with R and Hadoop, SPD 2013.

Reference Books

1. Tom White, "Hadoop: The Definitive Guide", Third Edition, O'Reilley, 2012.
2. Eric Sammer, "Hadoop Operations", O'Reilley, 2012.
3. E. Capriolo, D. Wampler, and J. Rutherglen, "Programming Hive", O'Reilley, 2012.
4. Lars George, "HBase: The Definitive Guide", O'Reilley, 2011.

Recommended by Board of Studies	5-3-2016		
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE2012	Software Security	L	T	P	J	C
		2	0	0	4	3
Pre-requisite	**SWE1701**	**Syllabus version**				
		v 1.0				

Course Objectives:

1. To understand concepts of software securities and insecurities.
2. To understand the requirement engineering for secure software and secure software design.
3. To analyse the types of software security testing techniques.

Expected Course Outcome:

1. Identify common security threats, risks, and attack vectors for software systems.
2. Formulate security goals of an information system, pointing out contradictory goals and suggesting compromises.
3. Evaluate security best practices and defense mechanisms for current software systems.
4. Enumerate limitations of existing defense mechanisms and alternatives to overcome them.
5. Apply contemporary formal mathematical modelling techniques to model and analyse the security of a software system.
6. Understand security protocols and verification issues.
7. Understand malicious code and other vulnerabilities along with mitigation mechanisims.
8. Understand and model the economics of cybersecurity.

Module:1	**Security issues in Software**	**6 hours**

Introduction, The problem, Software assurance and software security, Threats to software security, Sources of software insecurity, Benefits of detecting software security defects early, Managing secure software development, Properties of secure software.

Module:2	**Requirements Engineering for Secure Software**	**7 hours**

The SQUARE process model: Identifying security requirements using the security quality requirements engineering (SQUARE) method, SQUARE sample outputs, Requirements elicitation, Requirements Prioritization

Module:3	**Secure Software Architecture and Design**	**7 hours**

Introduction, Software security practices for architecture and design: Architectural risk analysis. Software security knowledge for architecture and design: Security principles, Security guidelines, and Attack patterns.

Module:4	**Secure Coding and Testing**	**6 hours**
Introduction, Code analysis, Coding practices, Software security testing, Security testing considerations throughout the SD.		
Module:5	**Security and Complexity**	**6 hours**
Security Failures, Functional and Attacker Perspective for Security Analysis, System Complexity Drivers and Security, Problem complexity		
Module:6	**Governance and Security**	**5 hours**
Security Governance, Characteristics of Effective Security Governance, Adopting an Enterprise software security Framework		
Module:7	**Managing a Secure Software**	**6 hours**
Security and Project Management – Project Scope and Plan, Resource, Estimate the Resources, Product and Project Resources, Measuring Software Security, Maturity of Practice.		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)				
1.	Julia H.Allen, Sean Barnum, Robert J.Ellison, Gary Mc.Graw, Nancy R.Mead, Software Security Engineering : A Guide for Project Managers, Addison-Wesley, 2011.			
Reference Books				
1.	Gary Mc.Graw, Software Security: Building Security, First Edition, Addison-Wesley , 2008.			
	Recommended by Board of Studies		5-3-2016	
	Approved by Academic Council	No. 40th	Date	18-3-2016

SWE2013	Advanced Java Programming	L	T	P	J	C
		3	0	0	4	4
Pre-requisite	**SWE1007**	**Syllabus version**				
		v. 1.0				

Course Objectives:

1. To understand java server side programming using Servlets, JSP and JDBC
2. To introduce the advanced java frameworks for improving the design

Expected Course Outcomes:

Upon completion of this course, the students will be able to

1. Understand and implement advanced-core Java concepts
2. Develop Java based Web applications using Servlets and JSP
3. Incorporate cutting-edge frameworks for improving the code design
4. To understand MVC framework, IoC and struts framework
5. Understanding application development using JSF
6. Understanding JSF navigational and event model
7. Understanding ORM and Hibernate

Module:1	Exploring Core Java	6 hours

Java Autoboxing and Annotations, Generics, Collections Framework, Concurrent Programming, Java NIO, Reflection, RMI

Module:2	Introducing JavaEE	6 hours

Enterprise Java, Basic Application Structure, Using Web Containers, Creating Servlets, Configuring Servlets, Understanding HTTP methods, Using Parameters and Accepting Form Submissions, Using Init parameters, File Uploading, Accessing Databases with JDBC

Module:3	Java Server Pages(JSP)	7 hours

Creating JSPs, Using Java within JSP, Combining Servlets and JSPs, Maintaining State using Sessions, JSP 2.0 EL, Using Javabeans components in JSP Documents, JSP Custom Tag Library, Integrating Servlets and JSP: Model View Controller Architecture

Module:4	MVC Frameworks	7 hours

Spring Framework: Understanding Inversion of Control (IoC), Aspect Oriented Programming (AOP) and Dependency Injection, MVC pattern for Web Applications, Spring Framework, Understanding Application Context, Bootstrapping Spring framework, Configuring Spring framework, **Struts Framework:** Introduction to Struts – Building a Simple Struts Application – Understanding Model, View and Controller Layer

Module:5	Java Server Faces(JSF)	6 hours

Introduction to Java Server Faces (JSF)- JSF Application Architecture – Building a simple JSF Application - JSF Request Processing Lifecycle – The Facelets View Declaration Language – Managed Beans and JSF Expression Language

Module:6	**JSF Navigation Model**	**5 hours**
JSF Navigation Model – User Interface Component Model – Converting and Validating data – JSF Event Model		
Module:7	**ORM and Hibernate**	**6 hours**
Data Persistence, Object/relational Mapping, Hibernate ORM, Mapping Entities to Tables		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1. Nicholas S. Williams, Professional Java for Web Applications, Wrox Press, 2014

Reference Books

1. Herbert Schildt, The Complete Reference-Java, Tata Mcgraw-Hill Eighth Edition, 2011
2. Ed Burns, Chris Schalk, JavaServer Faces 2.0, The Complete Reference, 2010, McGraw-Hill Publishers
3. Christian Bauer, Gavin King, Gary Gregory, Java Persistence with Hibernate, 2015
4. Craig Walls, Spring in Action Paperback , Manning Publications, 2014
5. James Holmes, Struts, The Complete Reference, 2007, McGraw-Hill Publishers

	Recommended by Board of Studies		5-3-2016	
	Approved by Academic Council	No. 40th	Date	18-3-2016

SWE2014	Advanced DBMS	L	T	P	J	C
		2	0	2	0	3
Pre-requisite	**SWE1004**	**Syllabus version**				
		v. 1.0				

Course Objectives:

1. To understand database design, tuning and queries.
2. To acquire knowledge on parallel and distributed databases and its applications.
3. To study the usage and applications of object oriented database
4. To understand the principles of intelligent databases.
5. To learn emerging databases such as XML, mobile databases.

Expected Course Outcome:

1. Comprehend the advanced features of databases.
2. Realize Database tuning
3. Design parallel and distributed databases.
4. Implement the concept of distributed transactions incorporating the Concurrency control mechanism.
5. Model and represent the real world data using object oriented database.
6. Embed the rule set in the database to implement intelligent database.
7. Design and Implement the XML data model

Module:1	Database Design And Tuning	5 hours

Introduction to physical database design – Guideline for index selection- Overview of database tuning – Conceptual schema tuning – Queries and view tuning

Module:2	Parallel and Distributed Databases	5 hours

Centralized and Client-Server Architectures – Server System Architectures - I/O Parallelism – Inter and Intra Query Parallelism – Inter and Intra operation Parallelism – Design of Parallel Systems- Distributed Architecture - Distributed Data Storage – Distributed Transactions – Commit Protocols – Concurrency Control – Distributed Query Processing – Case Studies

Module:3	Object Databases:	6 hours

Objects Identity – Object structure – Type Constructors – Encapsulation of Operations – Methods – Persistence – Type and Class Hierarchies – Inheritance – Complex Objects – Object Relational features- ODMG Model – ODL – OQL

Module:4	Active Databases:	6 hours

Syntax and Semantics - Taxonomy- Applications-Design Principles for Active Rules- Temporal Databases: Overview of Temporal Databases- TSQL2

Module:5	Deductive and XML Databases	6 hours

Logic of Query Languages – Datalog- Recursive Rules-Syntax and Semantics of Datalog Languages- Implementation of Rules and Recursion- Recursive Queries in SQL- XML Data Model –XML Documents- DTD - XML Schema - XML Querying.

Module:6	**Contemporary issues** .	**2 hours**
	Total Lecture hours:	**30 hours**

Text Book(s)

1. Elmasri, S.B. Navathe, ‑Fundamentals of Database Systems‖, 2011, Sixth Edition, Pearson Education/Addison Wesley.

Reference Books

1. Henry F Korth, Abraham Silberschatz, S. Sudharshan, ‑Database System Concepts‖, Sixth Edition, McGraw Hill, 2011.
2. Thomas Cannolly and Carolyn Begg, ‑Database Systems, A Practical Approach to Design, Implementation and Management‖, Sixth Edition, Pearson Education, 2014.
3. C.J.Date, A.Kannan, S.Swamynathan, ‑An Introduction to Database Systems‖, Eighth Edition, Pearson Education.2006.
4. G.K.Gupta,‖Database Management Systems‖, Tata McGraw Hill, 2011.

List of Challenging Experiments (Indicative)

1.	Creation of Tables , Views, Synonyms, Sequence, Indexes, Save point a. Creating an Employee database to set various constraints and writing SQL queries to retrieve information from the database. b. Performing Insertion, Deletion, Modifying, Altering, Updating and Viewing records based on conditions. c. Creation of Views, Synonyms, Sequence, Indexes, Save point.	
2.	Query Processing – Implementation of an efficient query optimizer Implement Query Optimizer with Relational Algebraic expression construction and execution plan generation for choosing an efficient execution strategy for processing the given query. Also design employee database and test the algorithm with following sample queries. a) Select empid, empname from employee where experience > 5 b) Find all managers working at London Branch	
3.	Parallel queries. Consider the application for VIT University Counselling. The campus, department and vacancy details are maintained in 3 sites. Students are allocated campus in these 3 sites simultaneously. Implement this application using parallel database [State any assumptions you have made]	
4.	Creating Database Link, executing distributed queries There are 5 processors working in a parallel environment and producing output. The output record contains campus details and students mark information. Implement parallel join and parallel sort algorithms to get the marks from different campus of the university and publish 10 ranks for each discipline.	
5.	Creating type,varray, nested table and querying it A University wants to track persons associated with them. A person can be an Employee or Student. Employees are Faculty, Technicians and Project associates. Students are Full time students, Part time students and	

	Teaching Assistants. Design an Enhanced Entity Relationship (EER) Model for university database. Write OQL for the following 5. Insert details in each object. 6. Display the Employee details. 7. Display Student Details. 8. Modify person details. Delete person details.	
6.	Active Databases Extend the design of university database by incorporating the following information. Students are registering for courses which are handled by instructor researchers (graduate students). Faculties are advisors to graduate students. Instructor researchers‘ class is a category with super class of faculty and graduate students. Faculties are having sponsored research projects with a grant supporting instruction researchers. Grants are sanctioned by different agencies. Faculty belongs to different departments. Department is chaired by a faculty. Implement for the Insertion and Display of details in each class.	
7.	Deductive Database Create triggers and assertions for Bank database handling deposits and loan and admission database handling seat allocation and vacancy position. Design the above relational database schema and implement the following triggers and assertions. 7. When a deposit is made by a customer, create a trigger for updating customers account and bank account 8. When a loan is issued to the customer, create a trigger for updating customer‘s loan account and bank account. 9. Create assertion for bank database so that the total loan amount does not exceed the total balance in the bank. When an admission is made, create a trigger for updating the seat allocation details and vacancy position.	
8.	Designing XML Schema and querying it. Construct a knowledge database for kinship domain (family relations) with facts. Extract the following relations using rules. Parent, Sibling, Brother, Sister, Child, Daughter, Son, Spouse, Wife, husband, Grandparent, Grandchild, Cousin, Aunt and Uncle.	
9.	Design XML Schema for the given company database Department (deptName, deptNo, deptManagerSSN, deptManagerStartDate, deptLocation) Employee (empName, empSSN, empSex, empSalary, empBirthDate, empDeptNo, empSupervisorSSN, empAddress, empWorksOn) Project (projName, projNo, projLocation, projDeptNo, projWorker) Implement the following queries using XQuery and XPath • Retrieve the department name, manager name, and manager salary for every department‘ • Retrieve the employee name, supervisor name and employee salary for each employee who works in the Research Department. • Retrieve the project name, controlling department name, number of employees and total hours worked per week on the	

	project for each project. • Retrieve the project name, controlling department name, number of employees and total hours worked per week on the project for each project with more than one employee working on it	
10.	Implement a storage structure for storing XML database and test with the above schema.	
	Total Laboratory Hours	30 hours

Recommended by Board of Studies	5-3-2016		
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE2015	Mainframe Computing	L	T	P	J	C
		3	**0**	0	0	**3**
Pre-requisite	**SWE1004**	**Syllabus version**				
		v. 1.0				

Course Objectives:

1. To understand the basic concepts of mainframe technologies.
2. To learn Mainframe programming Language.

Expected Course Outcome:

1. Understand Mainframe hardware
2. Understand Mainframe operating system
3. Develop mainframe applications
4. Explore concepts in Job Control Language and its associated programs
5. Understand basic concepts in COBOL programming
6. Practice problem solving in File Processing and Table Processing in COBOL Programming
7. Learn and explore basic concepts in DB2 and practice queries using DB2
8. To design interactive application based systems using TSO/ISPF

Module:1	**Evolution of Mainframe hardware**	**5 hours**

Overview of Computer Architecture -Classification of Computers - micro, mini, mainframes and super computer – Mainframe computer - key features - benefits - Evolution of Mainframes - Different hardware systems. Mainframes OS and Terminology: Operating systems on mainframes, Batch processing vs. online processing – mainframe operating system - evolution - concepts of Address space, Buffer management - Virtual storage - paging - swapping – Dataset management in mainframes**.**

Module:2	**z/OS and its features**	**4 hours**

Z-operating system (Z/OS) - Virtual storage - Paging process -storage Managers - Program execution modes - Address space - Multiple virtual system(MVS) , MVS address space, Z/OS address space - Dataset - sequential and partial dataset - Direct access storage device(DASD) - Access methods - Record formats -Introduction to virtual storage access methods(VSAM) - Catalog – VTOC

Module:3	**Introduction to JCL**	**5 hours**

Introduction to Job Control language - Job processing – structure of JCL statements - Various statements in JCL - JOB statement - EXEC statement – DD statement - JCL procedures and IBM utility programs.

Module:4	**COBOL Programming 1**	**7 hours**

Introduction – History, evolution and Features, COBOL program Structure, steps in executing COBOL. Language Fundamentals – Divisions, sections, paragraphs, sections, sentences and statements, character set, literals, words, figurative constants, rules for forming user defined words, COBOL coding sheet.. Data division – Data names, level numbers, PIC and VALUE

clause, REDEIFNES, RENAMES and USAGE clause. Procedure Division – Input / Output verbs, INITIALIZE verb, data movement verbs, arithmetic verbs, sequence control verbs.

Module:5	**COBOL Programming 2**	**8 hours**

File processing – Field, physical / logical records, file, file organization (sequential, indexed and relative) and access mode, FILE-CONTROL paragraph, FILE SECTION, file operations. File handling verbs – OPEN, READ, WRITE, REWRITE, CLOSE. Table processing – Definition, declaration, accessing elements, subscript and index, SET statement, SEARCH verb, SEARCH ALL verb, comparison. Miscellaneous verbs – COPY, CALL, SORT, MERGE, STRING, UNSTRING verbs.

Module:6	**Overview of DB2**	**7 hours**

Introduction to DB2 – System Service component, Database Service component, Locking Service component, Distributed Data Facility Services component, Stored Procedure component, catalogs and optimizer. DB2 Objects and Data Types -DB2 Objects Hierarchy, Storage groups, Database, Table space, Table, Index, Synonyms and aliases, Views, Data Types. DB2 SQL programming – Types of SQL statements, DCL, DDL, DML, SPUFI utility. Embedded SQL programming – Host variable, DECLGEN utility, SQLCA, single/multiple row manipulation, cursors, and scrollable cursors

Module:7	**Interactivity using TSO/ISPF**	**7 hours**

Key TSO Concepts-The Two Commandments of TSO Logging On to TSO-SPF Initialization and Invocation-Keyboard-Allocating a Data Set-Creating (Editing) a Program Data Set-Printing a Data Set-Running a Program Viewing and Printing Program Results-Compressing a Partitioned Data Set-TSO Initialization-Logging Off of TSO

Module:8	**Contemporary issues**	**2 hours**

	Total Lecture hours:	**45 hours**

Text Book(s)

1. M.Ebbers., John Kettner , Wayne O‘Brien , Bill Ogden, ―Introduction to the new mainframe: z/OS basics‖ , March 29, 2011, third edition , Vervante.

Reference Books

1. Craig S. Mullins,DB2 Developer‘s Guide: A Solutions-Oriented Approach to Learning the Foundation and Capabilities of DB2 for z/OS , March 2012 , (6th Edition) IBM Press
2. Stern,Stern and Ley.,‖COBOL for the 21st Century‖ , 2013 ,11th Edition, Wiley.

	Recommended by Board of Studies		5-3-2016	
	Approved by Academic Council	No. 40th	Date	18-3-2016

SWE2016	Semantic Web Technologies	L	T	P	J	C
		3	0	0	0	3
Pre-requisite	**SWE1008**	**Syllabus version**				
		v. 1.0				

Course Objectives:

1. To understand the need of Semantic Web Technologies
2. To know the methods to discover, classify and build ontology for more reasonable results in searching.
3. To build and implement a small ontology that is semantically descriptive of chosen problem domain.
4. To implement applications that can access, use and manipulate the ontology.

Expected Course Outcome:

1. To understand the need of semantic web technologies
2. To know the methods to discover, classify and build ontology for reasonable results in searching
3. Implement the Programs using XML, RDF and OWL
4. To build and implement a small ontology that is semantically descriptive of chosen problem domain
5. Understand logics, semantics and reasoning and implement writing rules
6. To implement applications that can access use and manipulate the ontology

Module:1	Introduction	4 hours

Introduction to the Syntactic web and semantic Web, Evolution of the Web, The visual and Syntactic Web, Levels of Semantics- Metadata for web information.

Module:2	Semantic Technologies	5 hours

Semantic web architecture and Technologies, Contrasting Semantic with Conventional Technologies, Semantic Modelling, and Potential of Semantic web Solutions and challenges of adoption.

Module:3	Ontological Engineering	5 hours

Ontologies, Taxonomies, Topic Maps – Classifying Ontologies- Terminological aspects: concepts, terms, relations between them, Complex Objects, Subclasses and Sub-properties definition, Upper Ontologies, Quality-Uses

Module:4	Resources For Ontology Building	6 hours

Methods and methodologies for building ontologies, Multilingual Ontologies, Ontology Development process and Life Cycle – Methods for Ontology Learning – Ontology Evolution – Versioning.

Module:5	**Structuring And Describing Web Resources**	**8 hours**
Structured Web Documents, XML, Structuring, Namespaces, Addressing, Querying, Processing, RDF, RDF Data Model, Serialization Formats – RDF Vocabulary – Inferencing – RDFS, Basic Idea, Classes, Properties, Utility Properties, RDFS Modeling for Combinations and Patterns – Transitivity		
Module:6	**Web Ontology Language**	**8 hours**
OWL-Sub-Languages, Basic Notations, Classes, Defining and Using Properties, Domain and Range – Describing Properties, Data Types, Counting and Sets, Negative Property Assertions, Advanced Class Description, Equivalence – Owl Logic.		
Module:7	**Semantic Web Tools**	**7 hours**
Development Tools for Semantic Web-Jena Framework, SPARL Queryinig SemanticWeb, Semantic Wikis, Semantic Web Services, Agent System, Conversion Tools, Graph Share Point Tools.		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1.	Breitman, Karin, Casanova, MarcoAntonio Truszkowski Walt: Semantic Web: Concepts Technologies and Applications 2014.

Reference Books

1.	Liyang Yu, –A Developer‘s Guide to the Semantic Web‖, Springer, First Edition, 2011
2.	John Hebeler, Matthew Fisher, Ryan Blace and Andrew Perez-Lopez, –Semantic Web Programming‖, Wiley, First Edition 2009.
3.	Dean Allemang and James Hendler, –Semantic Web for the Working Ontologist: Effective Modeling in RDFS and OWL, Morgan Kaufmann‖, Second Edition 2011.

Recommended by Board of Studies		5-3-2016	
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE2017	Parallel Programming	L	T	P	J	C
		3	0	2	0	4
Pre-requisite	**SWE1007**	**Syllabus version**				
		v. 1.0				

Course Objectives:

1. To learn to develop parallel algorithms and map them with processor architectures
2. To understand the parallelization of basic mathematical and engineering algorithms
3. To learn contemporary parallel architectures and programming

Expected Course Outcome:

1. Understand basic parallel architectures and parallel programming concepts
2. Learn parallel programming languages for Symmetric Shared Memory Systems
3. Learn parallel programming languages for distributed shared memory systems
4. Develop algorithms for specific parallel architectures
5. Develop efficient parallel algorithms for sorting problem
6. Learn parallelization techniques for image processing algorithms
7. Develop efficient parallel algorithms for optimization problems

Module:1	**PRAM ALGORITHMS**	**8 hours**

Introduction to Parallel Programming - Flynn‘s Taxonomy-PRAM model of parallel computation - EREW-CREW-CRCW- Mapping theorem -Parallel reduction – prefix sums – list ranking – preorder tree traversal – merging two sorted lists – graph coloring

Module:2	**SHARED MEMORY PROGRAMMING**	**6 hours**

Shared-memory model – OpenMP standard – parallel for loops – parallel for pragma – private variables – critical sections – reductions – parallel loop optimizations – general data parallelism – functional parallelism – case studies: the sieve of Eratosthenes, Floyd's algorithm, matrix-vector multiplication – distributed shared-memory programming – DSM primitives

Module:3	**MESSAGE-PASSING PROGRAMMING**	**5 hours**

The message-passing model – the message-passing interface – MPI standard–basic concepts of MPI: MPI_Init, MPI_Comm_size, MPI_Comm_rank, MPI_Send, MPI_Recv, MPI_Finalize –timing the MPI programs: MPI_Wtime, MPI_Wtick – collective communication: MPI_Reduce, MPI_Barrier, MPI_Bcast, MPI_Gather, MPI_Scatter – case studies: the sieve of Eratosthenes, Floyd's algorithm, Matrix-vector multiplication

Module:4	**PARALLEL MATRIX MULTIPLICATION ALGORITHMS**	**6 hours**

Matrix multiplication on 2D Mesh SIMD model – Related theorems -Hypercube SIMD model – shuffle exchange SIMD model – UMA Multiprocessor – Block matrix multiplication – Algorithms for multicomputer – Row-column and block oriented algorithms.

Module:5	**PARALLEL SORTING ALGORITHMS**	**6 hours**

Enumeration sort – Lower bounds on Parallel sorting – Odd Even Transposition sort – Bitonic

merge – sequence – Bitonic merge on shuffle exchange network – two dimensional mesh network – Hypercube network – Parallel quicksort – Hyperquick sort		
Module:6	**PARALLELIZATON OF IMAGE PROCESSING ALGORITHMS**	**5 hours**
Low-Level Image Processing – Point Processing – Histogram – Smoothing, Sharpening and Noise Reduction – Edge Detection – The Hough Transform – Transformation into the frequency domain		
Module:7	**PARALLELIZATION OF SEARCHIING AND OPTIMIZATION**	**7 hours**
Applications and Techniques – Branch and Bound Search – Genetic Algorithms – Successive Refinement – Hill Climbing.		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1. Michael J. Quinn, Parallel computing theory and practice, Second Edition, McGraw Hill, 2012.

Reference Books

1. B. Wilkinson and M. Allen, Parallel Programming – Techniques and applications using Networked workstations and parallel computers, Second Edition, Pearson Education, 2005.
2. Michael J. Quinn, Parallel Programming in C with MPI and OpenMP, McGraw-Hill Higher Education, 2003
3. Ananth Grama, Anshul Gupta, George Karypis, Vipin Kumar, Introduction to Parallel Computing, 2/E, Addison Wesley, 2003.
4. David B. Kirk, Wen-mei W. Hwu, Programming Massively Parallel Processors: A Hands-on Approach, MK Publishers, 2010

List of Challenging Experiments (Indicative)

1.	Implement the following using 2. Shared Memory model [Low Level] 3. Message Passing model [Medium Level] 4. CUDA Programming model [High Level]	
2.	Write parallel programs to solve Laplace‘s equation using each of the following three ways: 2. Standard Jacobi Iteration 3. Red-black Iteration 4. Multigrid Jacobi Iteration Use a 256 X 256 mesh of points initialized along the four edges to 10.0, 5.0, 10.0 and 5.0. Stop iterations when the differences between iteration values are all less than 0.01. Use 16 processes. For the standard and red-black	

	iteration methods, partition the problem into 16 columns of 16X256 points each, one column for each of the 16 processes. For the multigrid iteration, start with a grid size of 16X16 and increase the grid size by a factor of 2, for every 10 iterations until the maximum grid size is reached. Continue iterations until the solution is obtained.	
3.	Write a parallel program to solve the room temperature distribution problem but by the direct means of Gaussian Elimination and back substitution rather than by iteration. Only the Gaussian elimination need be computed in parallel; the back substitution may be done on one processor. First, determine the elements of the array A of the system of linear equations, Ax=0. Since this array will always have nonzero elements along the diagonal, partial pivoting should be unnecessary. Next, decompose the problem so that 10 consecutive rows are handled by one process.	
4.	You have been commissioned by a major film studio to develop a really fast ‒morphing‖ package that will change one image into another image. You come up with the idea of having two images, the original image and the final image, and changing each pixel on the original image to become closer and closer to the pixels of the final image in a lock-step SIMD fashion. This method is certainly embarrassingly parallel, although it may not create a very smoothly changing shape. Experiment with the method and demonstrate it to the studio using pictures of actors.	
5.	NASA has given you the task of writing a really fast image-recognition program, fast enough that a Venusian CAT (Commercial Access Transport) is able to capture touchdown sites from topographic images made by the VERMIN satellite while passing over the mapped area at a speed of 1000 km/hour. The VERMIN image maps are of a 5 Km X 5 Km area and have 0.5m resolution both horizontally and in altitude. Appropriate landing sites are areas in which there is a 1.5m maximum altitude variation within a 25m circle. Create sample image maps of imperfect terrain.	
6.	A Nationwide parcel delivery company, is reassessing the placement of its hubs that collect and distribute parcels. Ideally, the hubs should be situated at strategic places across the country to minimize costs and delivery times. You have been commissioned to make a study of possible alternative sites for the hubs and decide to write a parallel program based on genetic algorithms. You assume that the number of parcels being received is directly proportional to the population, and for a first approximation only the major cities are considered. Write the program, developing suitable input data and constraints. One constraint is the number of hubs.	
7.	A recently discovered planetoid, Geometrica, has a most unusual surface. By all available observations the surface can be modeled by the formula $H=35{,}000\sin(3\Theta)\sin(2\rho)+9700\cos(10\Theta)\cos(2\rho)-800\sin(25\Theta+0.03\pi)+550\cos(\rho+0.2\pi)$ Where H is the height above or below sea level, Θ is the angle in the equatorial plane and ρ is the angle in the polar plane. Write an embarrassingly parallel program to use hill climbing to find the (Θ,ρ)	

	position of the highest point above sea level on Geometrica‘s surface.			
			Total Laboratory Hours	30 hours
Recommended by Board of Studies	5-3-2016			
Approved by Academic Council	No. 40th	Date	18-3-2016	

SWE2018	Object Oriented Analysis and Design	L	T	P	J	C
		3	0	2	0	4
Pre-requisite	SWE1701	Syllabus version				
		v.1.0				

Course Objectives:

1. To explore designing interface objects for real life applications.
2. To prepare a model with object oriented approach that transforms into implementation specific drafts.
3. To analyze and design the requirements of software development using UML

Expected Course Outcome:

1. Understand basic concepts of object oriented approach through unified process.
2. Compareherd software development life cycle through object oriented approach
3. Recognize the object modeling and emerging phases of UML
4. Apply UML with static and dynamic behaviour for an interactive design process.
5. Apply UML by mapping analysis and design to software components
6. Identify the roles of classes and various relationships associated with the objects
7. Create classes as per object oriented design principles, standards and guidelines.
8. Transform identified analysis into design form which maps to implementatio in on real-life applications

Module:1	Introduction	6 hours

Structure of Complex Systems, Decomposing Complexity - Elements of Analysis and Design, Object Modeling - Unified Process - Phases of Unified Process –Benefits and Risks of Object Oriented Development.

Module:2	Object Oriented System Design	6 hours

Object Oriented Systems Development Life Cycle. Macro and Micro Process Development. Discussion on few Examples of OOAD Application Scenarios-Choosing a case study for OOAD.

Module:3	Methodology Modeling	6 hours

Object Oriented Methodologies-Rumbaugh et al.‘s object modeling technique-The Booch Methodology-The Jacobson et al. Methodologies.

Module:4	Design using UML Diagrams	6 hours

Introduction to UML as an Analysis and Design Tool, Class Diagrams, State Transition Diagrams, Object Diagrams, Interaction Diagrams, Use case Diagrams, Activity Diagrams, Collaboration Diagrams and Module Diagrams-Case Studies.

Module:5	**Implementation Diagrams**	**6 hours**
Component Diagram, Deployment Diagrams – Mapping of Diagrams to Analysis and Design Components		
Module:6	**Object Oriented Analysis**	**6 hours**
Identifying use cases - Object Analysis - Classification – Identifying Object relationships - Attributes and Methods.		
Module:7	**Object Oriented Analysis**	**7 hours**
Identifying use cases - Object Analysis - Classification – Identifying Object relationships - Attributes and Methods.		
Module:8	**Contemporary issues** .	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1. Ali Bahrami,‖Object Oriented System Development‖, Tata McGraw-Hill, 2012.

Reference Books

1. Grady Booch, Robert A. Maksimchuk , Michael W. Engle, Bobbi J. Young, Jim Conallen , Kelli A. Houston, ‒Object Oriented Analysis and Design with Application‖ ,3rd edition, Addison Wesley, 2011.
2. Schach and Stephen R., "An Introduction to Object-Oriented Systems Analysis and Design with UML and the Unified Process", Tata McGraw Hill, 2003.
3. Charles Richter, ‒Designing Flexible Object-Oriented Systems with UML‖, Techmedia, 2000.
4. Grady Booch, Ivar Jacobson, James Rumbaugh, The Unified Modelling Language User Guide, Second Edition, Pearson, 2012

List of Challenging Experiments (Indicative)

1. **STUDENT MARK ANALYSIS SYSTEM**
The XYZ University has decided to provide web-based student mark analysis system for the students in different Engineering colleges. The University maintains a database which contains student academic details belonging to various colleges. Colleges have various departments and each department has at most 4 sets of students studying in different semesters. If the particular semester students have got 2 sections then totally 8 class counselors are in charge for those classes. Likewise, in each and every department and colleges, there will be a set of class counselors who will operate the Student Mark Analysis System and have the University correspondence. The student will have maximum of 6 theory subjects and 2

	practical subjects in each semester. Each subject is evaluated for 100 out of which 20 marks for internals and 80 for external. The class counselor‘s responsibility is to put internal marks out of 20 and collects the external marks which are out of 80 from university after central valuation through university exam correspondent of the college. The class counselor analyses the marks got by the student in every subject based on the criteria. He/She calculates the overall pass percentage of the class and also department overall percentage is calculated. From each department overall percentage, the overall performance of the college is fetched. Based on some criteria, department wise 3 well performed students in every semester are identified and honored. The students can logon to the specified website and can view his/her report card. The students can also apply for revaluation by downloading appropriate form and filling up the details. He/She can send it to the university through university exam correspondent by attaching the printed revaluation form and Demand Draft for the specified amount. If there is a correction/no change in the mark, university will intimate through university exam correspondent. The class counselor then revises/updates the mark analysis that is done for specified class and corresponding details are updated.	
2.	QUIZ SYSTEM ABC Engineering college has decided to provide online quiz system to its student in various streams. The system will comprise of a database containing questions and answers on various subjects under various streams. The student will be able to login to the system through various desktop PCs available in the campus. The student will be able to take up a quiz view his/her performance over a period of time. The student will be able to take a break from a quiz only once for a maximum of 15 minutes. The quiz will comprise of only multiple choice questions. The duration of the quiz will be 1 hour and the student will be asked 50 questions. The system shall provide immediate feedback to the student whether he/she has passed or failed in the attempt based on the criteria after answering all questions. The individual lecturers will be responsible for generating the questions and answers for the question bank. The lecturers shall enter the moderated questions and answers in the question bank. The lecturers shall also manage their student‘s details in the system as part of which necessary login and password shall be created. The lecturers can view the performance history of their students and provide feedback to the student in the system. At the end of the semester the lecturers shall print a consolidated performance history of each student and provide it to their student with their comments	
3.	ON-LINE TICKET RESERVATION SYSTEM You have been asked to develop a web based ticket reservation system for the Southern Indian railways. The Southern Indian Railways has approximately around 300 major Railways Stations. The Railways wants to reduce the waiting time of the passengers by automating the ticket reservation process. The passenger will mention their reservation details by writing it in the reservation form to the ticket issuer. On receiving the reservation form from the passenger, the ticket issuer will enter the details in the online ticket issue form by retrieving necessary details from the central database. The ticket issuer checks whether the specified train has enough number of seats, births	

	and requested class in the specified date. Then if all the conditions are fine and if the passenger makes confirmation, the ticket issuer collects the money from the passenger and returns the tickets with relevant information printed on the tickets with balance amount if any. The ticket issuer updates in the central database. The ticket issuer also checks whether the passenger is a senior citizen. If he/she is so, then only 50% of the ticket cost is taken. If the passenger tickets are in waiting list he can keep track of his ticket number and can check his confirmation by browsing Indian railways web site. If the passenger wants to cancel the tickets within 24 hours of journey date his/her cancellation is made and 40% of the ticket cost is returned back. If he/she cancels the ticket before 24 hours of journey only 20% of ticket cost is taken and rest is returned. Then the system allots the cancelled seat/birth to the passenger who is at present in top position in the waiting lists and his/her seat/births confirmed.	
4.	PAYROLL SYSTEM Payroll system is the heart of any human resource system of an organization. The solution has to take care of the calculation of salary based on employee cadre, income tax calculation and various detection to be done from salary including statutory deduction like income tax and provident fund deduction. It has to generate pay slip, check summary and MIS reports. •Some employees work by the hour and they are paid in hourly rate. They submit daily time cards that record the date and number of hours work for the particular charge number. If someone works for more than 8 hours, the company pays them 1.5times their normal rate for those extra hours. •Some employees are paid a flat salary. Even though they are paid a flat salary, they submit daily time cards that record the date and hours worked. This is so the system can keep track of the hours worked against particular charge numbers. •Some of the salaried employees also receive a commission based on their sales. They submit purchase orders that reflect the date and amount of the sale. The commission rate is determined for each employee, and is one of 10%, 15%, 25% or 35% Employee Information to be maintained. 1. personal Information 2. Family Information 3. Qualification 4. Experience 5. Health Information 6. Bank Account 7. Company Information 8. Leave Eligibility 9. Salary Reports to be Generated 1. Pay slip 2. Department wise Salary 3. Employee wise Salary	
5.	COURSE REGISTRATION SYSTEM You have been asked to develop a new Course Registration System for your college. The college wants a web based system to replace its manual system. The college provides education in various streams. In any stream, the entire	

	registration is divided into semesters. The new system should allow the aspirants to submit their application online. Once their applications have been approved and they have been admitted into the college, the system should send an automatic welcome e-mail along with login id and password to the e-mail address of your students. The e-mail address is specified as part of an application. For students without any e-mail address, the system shall print the welcome letters to be posted. The students would also have selected their stream of interest. Each stream will have a set of courses which are mandatory, and a certain number of elective courses. These electives will applicable from the fifth semester onwards. The student has to select two electives. The complete list of courses is maintained in the database. This database belongs to another system and hence cannot be updated or changed by the new system in any manner. The database can only be read by the new system. The beginning of the semester, the head of the department will create necessary class and allocation of lectures to the classes for his department. The HOD may make changes in the allocation during the progress of the course. The system maintains the history of all the professor who has conducted a class throughout the semester. The lecturer will use the system to update the marks of the student (Project, Assignment, Internal Test Marks and the semester and the examination marks).the lecturer will also mark the attendance of the student in the system. The student can view his /her marks and attendance through the system. In addition to the above, the system also keeps track of residential status of the system. The student may be hosteled or a day scholar. If he is a hosteled, the system will maintain his/her hostels‘ name, room number and the fees pertaining to the same.	
		Total Laboratory Hours 30 hours

Recommended by Board of Studies	5-3-2016		
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE2019	Design Patterns	L	T	P	J	C
		2	0	0	4	3
Pre-requisite	**SWE1701**	**Syllabus version**				
		v.1.20				

Course Objectives:

1. To acquaint students with the basic of patterns, categories, and their usage.
2. To make the student understand the relation between OOPS paradigm and design patterns
3. To make the students understand how design patterns simplify the object creation process.
4. To make the students understand how design patterns simplify the structural rearrangement.

Expected Course Outcome:

1. Ability to understand the need for pattern, remember their types and significance.
2. Ability to understand the relation between OOPS paradigm and design patterns.
3. Ability to apply the suitable creational pattern for the object creation problem and evaluate their effectiveness.
4. Ability to apply the suitable structural pattern to make structural rearrangements.
5. Ability to apply the suitable behavioral pattern to provide special purpose for objects and analyze their interaction
6. Ability to analyze the usage of design patterns for industry scenarios.

Module:1	**Introduction**	**5 hours**

Introduction to patterns – Pattern categories – Relationship – Pattern description –Description of architectural patterns.

Module:2	**Design Pattern**	**5 hours**

Introduction – MVC, Describing Design Patterns -Problem solving by Design Pattern – Guidelines for selecting & using Design pattern.

Module:3	**Creational Pattern**	**6 hours**

Abstract factory – Builder – Factory methods – Prototype – Singleton – Real world examples.

Module:4	**Structural Pattern**	**6 hours**

Adapter – Bridge – Composite – Decorator – Real world example, Façade – Flyweight – Proxy – Real world examples.

Module:5	**Behavioral Pattern**	**6 hours**
Chain of responsibility – Command – Interpreter – Iterator – Mediator – Real world examples, Memento - Observer - State - Strategy – Template method – Visitor –Real world examples.		
Module:6	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**30 hours**

Text Book(s)

1.	Erich Gamma, Ralph Johnson, Richard Helm and John Vlissides, –Design Patterns: Elements of Reusable Object-Oriented Software‖, Pearson Education, 2015.

Reference Books

1. Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, Michael Stal, –Pattern-Oriented Software Architecture: A System of Patterns‖, Wiley India Pvt. Ltd., 2011.
2. Cay Horstmann, –Object-Oriented Design and Patterns‖, Wiley India Pvt. Ltd, 2012

Recommended by Board of Studies	12-8-2017		
Approved by Academic Council	No. 47th	Date	5-10-2017

SWE 2020	Software Metrics	L	T	P	J	C
		2	0	0	4	3
Pre-requisite	SWE1701	Syllabus version				
		v.1.20				

Course Objectives:

1. To understand data analysis metrics and models to assess software products.
2. To emphasize the use of software product and quality metrics.
3. To study various metrics models in the applications of software design and production

Expected Course Outcome:

1. To understand the challenges and difficulties of applying software metrics.
2. Choose appropriate metrics to collect data and use them to make predictions.
3. Carry out data analysis and visualization
4. Capture a key aspect of software size.
5. Capture a key aspect of software structure.
6. Identify a variety of quality models and evaluation techniques.
7. Make decisions for software project risk assessment and prediction.
8. Apply and evaluate the data analysis methods to validate decisions

Module:1	Basics of Measurement	5 hours

Measurement in Software Engineering - Scope and basics of Software Measurement - A Goal-Based Framework for Software Measurement- Applying the Framework - Software Measurement Validation.

Module:2	Software Metrics Data Collection	5 hours

Empirical Investigation-Principles of Empirical Studies-Planning Experiments-Planning Case Studies as Quasi-Experiments-Relevant and Meaningful Studies-Software Metrics Data Collection, Classical Data Analysis & Statistical Test

Module:3	Measuring Internal Product Size and Structure	6 hours

Measuring Internal Product Attributes: Size-Properties of Software Size-Code Size-Design Code-Requirement Analysis and Specification size-Functional size Measures and Estimators-Application of Size Measures-Problem, Solution size, Computation complexity-Tools for product Size Measurement.

Module:4	**External Product Attributes**	**6 hours**
Modeling Software Quality-Measuring Aspects of Quality-Usability--Maintainability -Security.		
Module:5	**Metrics for Decision Support**	**6 hours**
Metrics for Decision Support- from Correlation and Regression to Causal Models- Bayes theorem and Bayesian Networks-Applying Bayesian Networks to the Problem of Software Defects Prediction-Bayesian Networks for Software Project Risk Assessment and Prediction.		
Module:6	**Contemporary issues** .	**2 hours**
	Total Lecture hours:	**30 hours**

Text Book(s)

1. Norman Fenton, James Bieman, –Software Metrics: A Rigorous and Practical Approach‖, 3rd Edition, CRC Press, 2015.

Reference Books

1. Stephan H. Kan,‖ Metric and Models in Software Quality Engineering –, Second Edition, Pearson Education, 2015

Recommended by Board of Studies	12-8-2017		
Approved by Academic Council	No. 47th	Date	5-10-2017

SWE2021	Software Configuration Management	L	T	P	J	C
		3	0	0	0	3
Pre-requisite	SWE1701	Syllabus version				
		v.1.0				

Course Objectives:

1. To understand the concepts of software configuration Management.
2. To learn how to use various SCM functions.
3. To use the techniques in the real life project.

Expected Course Outcome:

1. Understand the basics of SCM and its functions
2. Understand the various types of defects and its classifications
3. Understand the various SCM Standards.
4. Understand of software process improvement models and to prepare a SCM plan
5. Understand how to organize people in the organizations and how to use right tool for right task
6. Understand how to implement SCM in the real life projects.
7. Understand the various implementation challenges and maintenance.
8. Apply the concepts to develop quality projects.

Module:1	Introduction to Software Configuration Management(SCM)	6 hours

Introduction-SCM and process improvement, Measurements, metrics and benefits. Configuration Identification, change control and auditing- implementation issues in SCM. Managing Roles. Preparing Project plan components for SCM.

Module:2	Configuration control & Auditing	6 hours

Configuration identification-impact, selection and acquisition. Configuration control-Defects classification, severity and preventions. Status auditing and automation and case studies.

Module:3	Advanced concepts in configuration verification and Audits	6 hours

Configuration verification and Audits, SCM: Advanced concepts, SCM: standards -military standards and International/commercial Standards

Module:4	Software process improvement models and SCM plans	6 hours

Introduction-CMMI, Information Technology Infrastructure Library (ITIL). Control Objectives for Information and Related Technology (COBIT). Software Engineering Body of Knowledge (SWEBOK). SCM plans-Tools and standards.

Module:5	**SCM organization and Tools**	**6 hours**
SCM organization- Automation and SCM team size, skill inventory database and CCB. SCM tools-Advantages, Implementation and functions of tools. Case studies on usage of various tools.		
Module:6	**SCM Implementation**	**6 hours**
Implementation-Plan, Risk, Strategies, Team and Performance measures. Different phases of SCM implementation. Source code repositories.		
Module:7	**SCM Implementation Challenges and Maintenance**	**7 hours**
Introduction- Implementation challenges. SCM operations and Maintenance; SCM Special circumstances. Case studies on SCM under Special circumstances.		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1. Alexis Leon, A Software configuration management handbook. Artech House. 2015.

Reference Books

1. Berczuk, S. P., & Appleton, B Software configuration management patterns: effective teamwork, practical integration. Addison-Wesley Longman Publishing Co., Inc..2011.
2. Mario E. Moreira, Software Configuration Management Implementation Roadmap, Wiley Publishers, Volume 1,2004.
3. Managing Global Software Projects, Gopalaswamy Ramesh, 2008, TMH.

Recommended by Board of Studies	5-3-2016		
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE 2022	Software Engineering Process, Tools & Methods	L	T	P	J	C
		2	0	0	4	3
Pre-requisite	SWE1701	Syllabus version				
		v.1.0				

Course Objectives:

1. To understand the process engineering meta-model and benefits of software and systems process engineering meta-model.
2. To know the fundamentals of software process improvement approaches and the capability maturity models with their levels.
3. To demonstrate the concepts of empirical studies and reporting experiments in software engineering.
4. To identify the applications of the software engineering process in industry point of view.

Expected Course Outcome:

1. Understand the software engineering process, models and improvements.
2. Identify suitable process improvement approach for any software organization.
3. Analyze the process measurement and experimental software engineering data.
4. Create status report and continuous improvement approaches with appraisals.
5. Perform process measurement and improvement with experimental data analyzes.
6. Summarise software engineering research in small and large scale industry.
7. Apply software engineering process methods and tools
8. Compare various kinds of process engineering tools and knowledge management.

Module:1	Introduction to Software Process Engineering	5 hours

Software Process Modeling and Improvement, Process Modeling Goals and Benefits, Prescriptive Process Model Classes, Product Line Engineering, Scaled Agile Framework, Process Standards, Process Representations in Organizations, Deploying Prescriptive Process Models

Module:2	Process Engineering Metamodel	5 hours

Goals of Descriptive Process Modeling, Creating a Descriptive Process Model, Criteria for Assessing Process Modeling Notations, Multi-view Process Modeling Language, Software & Systems Process Engineering Meta-model (SPEM 2.0)

Module:3	Process Improvement and Measurement	6 hours

Model-Based Improvement Approaches, CMMI, Maturity Levels, Categories of CMMI Processes, CMMI Process Areas, Components of CMMI Process Areas, SCAMPI Appraisals, Continuous

Improvement Approaches, Process Improvement and Measurement: The GQM and GQM+ Approach, Aligning Improvement Goals and Strategies with Business, Applying Measures to Process Management: Collecting and Retaining Data, Analyzing Data		
Module:4	**Empirical Studies :**	**6 hours**
Controlled Experiments: Research in the Small, Case Studies: Research in the Typical, Surveys: Research in the Large, Reporting Experiments in Software Engineering, Building Theories in Software Engineering Process Simulation: Software Process Simulation, Method for Developing Simulation Models, Plug & Play Process Models, Combining Process Simulation and Empirical Studies		
Module:5	**Process Engineering Tools & Knowledge Management:**	**6 hours**
Eclipse Process Framework – Composer, Create Method Content, Reuse Method Content, Work with Processes, Publish Method Content. Learning Modes & Knowledge Life Cycle, Knowledge in People, Teams & Organizations.		
Module:6	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**30 hours**

Text Book(s)	
1.	Jürgen Münch, Ove Armbrust, Martin Kowalczyk, Martín Soto-Software Process Definition and Management-Springer-Verlag Berlin Heidelberg, 2012
Reference Books	
1.	Gerard O'Regan – Introduction to Software Process Improvement - Springer-Verlag London Limited, 2011
2.	Kurt Schneider -Experience and Knowledge Management in Software Engineering-Springer-Verlag Berlin Heidelberg , 2009

Recommended by Board of Studies	5-3-2016		
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE2023	Automotive Software Engineering	L	T	P	J	C
		3	0	0	0	3
Pre-requisite	SWE1701	Syllabus version				
		v.1.0				

Course Objectives:

1. The main objective is to impact knowledge and understanding of the innovations in the automotive field to the application domains of software engineering

Expected Course Outcome:

1. Clear knowledge about problem solving skills in DS & Algorithms concepts.
2. Understand the Automotive System components and system architecture.
3. Understand the Real time system concepts and constraints
4. Applying suitable process Model, configuration management and project management technique for automotive system.
5. Managing automotive system Requirements and contract management.
6. Gathering automotive system user requirements and designing logical architecture of the system.
7. Designing overall automotive system architecture including data model and implementation.
8. Applying software testing techniques to automotive system using Eclipse.
9. Analyzing the Contemporary issues in Applications of automated software engineering in industry

Module:1	Overview of Automotive System:	6 hours

Driver-Vehicle Environment System – Operation, User Interface, Sensors and Actuators, Software Functions, Installation space, Variants and Scalability, System Architecture

Module:2	Software Engineering of System Basics:	6 hours

Control System, Discrete System, Embedded System, Real Time System, Distributed System and Networked Systems

Module:3	Support Process for Automotive Software Engineering:	6 hours

Process Model and Standards, Configuration Management, Project Management

Module:4	Subcontractor Management:	6 hours

Subcontractor Management, Requirement Management and Quality Assurance

Module:5	**Core Processes for Automotive Software Engineering:**	**7 hours**
User Requirements Analysis and Specification, Logical System Architecture and Specification, Software Component		
Module:6	**Methods for Development and Service**	**6 hours**
Design and Implementation of System Architecture, Software function, Data Model.		
Module:7	**Software Quality Testing Techniques & Services**	**6 hours**
Available techniques for Integration and Testing, Software Updates through Flash Programming, Debugging using Eclipse		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1.	Robert Oshana & Mark Kraeling, –Software Engineering for Embedded Systems: Methods, Practical Techniques, and Applications‖, 1st Edition, Newnes, 2013

Reference Books

1.	Ian Sommerville,Software Engineering, 9th Edition, Addision-Wesley, 2010
2.	William E. Lewis , –Software Testing and Continuous Quality Improvement‖, Third Edition, Auerbach Publications, 2008
3.	Jorg Schauffele, Thomas Zurawka, –Automotive Software Engineering: Principles, Processes, Methods, and Tools‖, SAE International, 2005

Recommended by Board of Studies	5-3-2016		
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE2024	Software Reuse	L	T	P	J	C
		3	0	0	0	3
Pre-requisite	**SWE1701**	**Syllabus version**				
		v.1.0				

Course Objectives:

1. To understand benefits and limitations of software reuse
2. To understand different ways of implementing software reuse.
3. To gain knowledge of design patterns and COTS techniques in the context of software reuse

Expected Course Outcome:

1. Analyze, implement and manage the reuse approach in the production environment.
2. Design a component with interfaces that adhere to standards.
3. Select and use a design pattern for the model.
4. Apply object oriented concepts to enable reuse.
5. Apply software reuse idea, architectural style and processes in their projects.
6. Apply software reuse in agile development methodology
7. Understand industry best practices in agile software development.

Module:1	Introduction	5 hours

Need - Success factors – Classical software reuse examples - Approach – Changes required in development environment and people to adopt reuse – Impact on business – Return On Investment (ROI) on reuse

Module:2	Reuse architecture	6 hours

Reuse architecture - Application Reuse - Component Reuse - Object and function Reuse – Layersof Reuse

Module:3	Adopting reuse	6 hours

Adopting organization for Reuse – Managerial responsibilities – People responsibilities – Setting up a process – Integration – Deployment

Module:4	OOPS and reuse	7 hours

Object oriented techniques for Reuse – Effect of reuse on using Encapsulation – Effect of reuse on using Modularization – Effect of reuse on using Inheritance.

Module:5	**Design patterns**	**7 hours**
Design patterns – Creational patterns – Structural patterns – Behavioral patterns – Case study		
Module:6	**CBT**	**6 hours**
Component based technology – Enterprise Java Beans – CORBA – ActiveX controls.		
Module:7	**Agile and reuse**	**6 hours**
Impact of reuse in agile development methodology – Legacy systems - Wrapping legacy software for reuse in SOA		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1. Erich Gamma, ―Design Patterns: Elements of Reusable Object-Oriented Software‖, Pearson Education, 2015.

Reference Books

1. Software Reuse: Methods, Models, Costs (2nd Edition),Ronald J.Leach, 2013, Aftermath publishers(ISBN-10:1939142350ISBN-13:978-1939142351)
2. Managing Software Reuse,Wayne C. Lim,2004, Prentice Hall (ISBN-10:0135523737 ISBN-13:978-135523735)
3. Ivar jacabson, Martin Griss, Patrick Hohson – Software Reuse. Architecture, Process and Organization for Business Success, Pearson Education, 2004.
4. Robert C. Martin, ―Agile Software Development, Principles, Patterns, and Practices‖, Pearson Education publishers, 2003.
5. Clemens Szyperski, ―Component Software: Beyond Object-Oriented Programming‖, Pearson Education publishers, 2003.

Recommended by Board of Studies	5-3-2016		
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE2025	Personal Software Process	L	T	P	J	C
		3	0	0	0	3
Pre-requisite	SWE1701	Syllabus version				
		v.1.0				

Course Objectives:

1. Elaborate an understanding of guidelines, principles, and theories behind PSP based approach for building software.
2. Focus on improving quality of software development in an individual level.
3. Carry out the steps to measure size, time, defects, and development process.
4. Emphasize to Manage quality and reduce defects in software projects.

Expected Course Outcome:

1. Understand the PSP -based approach for developing software
2. Plan for building and measuring the size of a product
3. Learn managing and scheduling a process.
4. Design the project plan for a software process
5. Understand the importance of software quality and techniques to identify defects in asoftware
6. Perceive the significance of Software Development Process and reduce the number ofdefects in their work
7. Manage the cost of quality and the personal commitment to quality

Module:1	An Overview of PSP and Time Management	6 hours

Software Engineering-Personal Software process-Improvement Process-Time Management-Logic of Time Management-Elements of Time Management-Categorizing Activities- Evaluating Time Distribution- Tracking Time-Recording time data- Handling Interruptions-Tracking Completed
Tasks.

Module:2	Product Planning and Product size measurement	5 hours

Product plan- Need for product planning- planning small jobs-job number log-cautions on using size measures-Program size- estimating program size-size measures in job number log-other size measures.

Module:3	Managing commitments and Schedules	7 hours

Defining commitment- Responsibly made commitments- Handling missed commitments- Importance of managing commitments- consequences of not managing commitments- ways to managing commitments- need for schedules- Gantt Chart- Making a project schedule- checkpoints- Tracking project plans- Tracking Earned value.

Module:4	Project plan and Software Development Process	6 hours

Need for project plans- Project plan summary- Time in phase- Use of processes- process script-

Checkpoint and phases- Updated project plan summary form- Planning Example.		
Module:5	**Defects and Software Quality**	**7 hours**
Defects-Importance of Software Quality- Increasing Risks of poor quality- Defects versus Bugs- Defect types- Understanding defects- Defect recording log- Steps in finding defects- ways to find and fix defects- Defect Removal time- Improving Defect removal rates- Reducing Defect injection rates.		
Module:6	**Product quality Management**	**6 hours**
Product quality- Testing-The Filter view of Testing- calculating yield values- Estimating the ultimate yield- Benefits of 100% process yield- Prototyping.		
Module:7	**Process Quality and Personal commitment to quality**	**6 hours**
Process measures-Defect Removal paradox- cost of quality- Appraisal/Failure ratio-Improving review rates- Making Commitment to quality- rewards of Accomplishment.		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1. Watts.S.Humphery, Introduction to the Personal Software Process, Pearson education, 2012.

Reference Books

1. Pomeroy-Huff,Marsha;Mullaney, Julia;Cannon, Robert; & Seburn, Mark, The Personal Software Process (PSP) Body of Knowledge, Version 1.0 (CMU/SEI-2005-SR-003). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2009.
2. Watts.S.Humphery, PSP: A Self- Improvement Process for Software Engineers,1st Edition, Addison Wesley Professional, 2005.
3. Software engineering Institute. Overview of Team Software Process and Personal Software process [Online]. Available URL:http://www.sei.cmu.edu/tsp/index.html (2008).

Recommended by Board of Studies	5-3-2016		
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE2026	Team Software Process	L	T	P	J	C
		3	0	0	0	3
Pre-requisite	SWE1701	Syllabus version				
		v.1.0				

Course Objectives:

1. To understand the benefits and potential problems of teaming, describing qualities and processes of effective teams, and describing the role of teamwork in system design.
2. To create a team charter to articulate how the team will track, manage and communicate project progress, changes in scope, changes in design, and defects.
3. To describe the quality assurance practices appropriate for each part of the development life cycle

Expected Course Outcome:

1. Analyze a problem, and identify and define the computing requirements appropriate to its solution.
2. Apply design and development principles in the construction of software systems of varying complexity.
3. Communicate effectively with a range of audiences, customers, supervisor, team mates, etc
4. Design the project plan for a software process
5. Develop a testing strategy ,plan for a software product
6. Understand the different roles in the software development team
7. Implement team software process for a software project

Module:1	TSP Overview	6 hours

TSP Overview - TSP principles, TSP Design, TSP Structure and Flow, TSP Process. Logic of the Team Software – Common Team Problems, Building Effective Teams.

Module:2	TSP Process	6 hours

Launching a Team Project – Team Goals, Team Member Goals, Role Goals, TSP Launch Scripts. Development Strategy – Conceptual Design, Risk Management, Reuse strategy, Strategy Scripts

Module:3	Development Plan	6 hours

Needs of Planning, Planning Process, Development plan Scripts, Quality Plan. Defining the requirements – Requirement changes, SRS, Requirement scripts.

Module:4	Design	6 hours

Designing with Teams – Design Principles, standards, designing for usability, testability, and

reuse, Design Reviews and Inspections, Design Scripts.		
Module:5	**Product implementation and Testing**	**6 hours**
Implementation standards and strategy , Review and Inspections, IMP Scripts, Testing Principles , Testing Strategy, Integration and system test strategy, Test Planning, Tracking and Measuring, Documentation		
Module:6	**Team Roles**	**6 hours**
Team Leader Role – Development Manger Role – Support Manger Role – Planning Manger Role -Quality Manager Role.		
Module:7	**Using TSP**	**7 hours**
Managing Yourself – Responsible, Defined Goals, Principles, Being on Team – Team work, communication among team members, Making and meeting commitments, Team activities, Team building , Accepting and Performing a Team Role, Building and Maintaining the Team		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)	
1.	Humphrey, Watts S., Introduction to the Team Software Process. Addison-Wesley, 2011
Reference Books	
1.	Humphrey, Watts S., TSP(sm): Leading Development Team, Pearson Education, 2010.

Recommended by Board of Studies	5-3-2016		
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE2027	Knowledge Management System	L	T	P	J	C
		2	0	0	4	3
Pre-requisite	**SWE1701**	**Syllabus version**				
		v.1.0				

Course Objectives:

1. To characterize knowledge and its creation, acquisition, representation, use and re-use and management.
2. To understand core concepts, methods, techniques and tools for computer support of knowledge management.
3. To design develop and integrate appropriate components and functions of various knowledge management systems.

Expected Course Outcome:

1. Understand Knowledge Management from the system perspective to the organizational perspective.
2. Identify key components of Knowledge Management foundations and supporting technology.
3. Distinguish among Knowledge Management Processes and corresponding systems.
4. Analyze the impacts of Knowledge Management on people, process, product and organization.
5. Characterize and design Knowledge capture systems based on different methodologies and technologies
6. Describe crucial requirement for Knowledge sharing systems and to suggest appropriate design.
7. Understand the design consideration for Knowledge discovery systems and identify appropriate techniques and tools
8. Assess and benchmark various Knowledge Management approaches

Module:1	**Introduction**	**4 hours**

What is Knowledge- Forces driving Knowledge Management- Knowledge Management System - Importance of Knowledge Management System- Issues in Knowledge Management, Principles of Knowledge Management

Module:2	**Processes and Systems**	**4 hours**

Knowledge Management processes- Knowledge Management Systems-Managing Knowledge Management Foundations-Application Exercises

Module:3	**Technologies, Systems and Organizational Impacts of Knowledge Management**	**6 hours**

Impact on People- Impact on Process- Impact on Product-Impact on Organizational Performance-

Technologies for Applying Knowledge-Developing Knowledge Application Systems-Types of Knowledge Application Systems		
Module:4	**Knowledge Capture Systems and Knowledge Sharing Systems**	**8 hours**
What are Knowledge capture systems?-Mechanisms for capturing Tacit knowledge using Organizational stories-Designing the knowledge capture systems-Concept Maps-Context-based Reasoning-Knowledge capture systems based on Context based Reasoning: What are Knowledge Sharing Systems – designing the Knowledge Sharing Systems-Barriers of Knowledge Sharing Systems-Specific types of Knowledge Sharing Systems-shortcoming of Knowledge sharing Systems-Knowledge Management Systems that share tacit Knowledge		
Module:5	**Knowledge Discovery Systems**	**6 hours**
Mechanisms for Knowledge Discovery-Technologies for Knowledge Discovery-Designing Knowledge Discovery System-Guidelines for employing Data mining techniques-Discovering Knowledge on the web		
Module:6	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**30 hours**

Text Book(s)

1. Irma Becerra-Fernandez and Rajiv Sabherwal, Knowledge Management Systems and Processes, Second Edition, Hardcover Import,Dec 2014

Reference Books

1. Chinmoy Mukherjee, ‒Knowledge Management, Engineering and Automation: Design, Implementation and Benefits of Knowledge Management ―, April 16, 2014.
2. KimizDalkir, Jay Liebowitz , ‒Knowledge Management in Theory and Practice‖, 2011.
3. Ronald Brachman, Hector Levesque ‒Knowledge Representation and Reasoning ‒, The
4. Morgan Kaufmann Series in Artificial Intelligence 2004
   John F. Sowa, ‒Knowledge Representation: Logical, Philosophical, and Computational Foundations‖, 2000.

Recommended by Board of Studies	5-3-2016		
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE2028	Software Engineering Economics	L	T	P	J	C
		3	0	0	0	3
Pre-requisite	SWE1701	Syllabus version				
		v.1.0				

Course Objectives:

1. Understand and able to apply the key software engineering economic fundamentals to real-world software economic issues
2. Illustrate through example the key software life cycle economics, including product and process life cycles; portfolios; proposals; investment decisions; pricing and costing, and earned value management.
3. Apply the concepts of risk and uncertainty to real-world software development projects, including goals; estimates; prioritization and decision making
4. Perform best-practice economic analysis methods
5. Understand software ecosystem

Expected Course Outcome:

1. An ability to understand the subject related concepts and contemporary issues
2. An ability to apply mathematics and science in engineering applications
3. An ability to solve social issues and engineering problems
4. To understand and apply the Macroeconomics and Microeconomic in advance
5. To apply and practice software Eco system
6. To identify contemporary issues in applying Software Cost Estimation techniques.
7. To apply the Earned value Management ,Performance Measurement, maintenance and challenges faced in software industry
8. An ability to use techniques, skills and modern engineering tools necessary for Software Engineering Economics practice

Module:1	Fundamentals of software economics	6 hours

Definitions-Economics, micro and macroeconomics, Economics and Software Engineering management, Finance, Accounting, Controlling, Cash flow, decision making process, inflation, depreciation, taxation, efficiency, time value of money, effectiveness, productivity

Module:2	Life Cycle Economics	5 hours

Product, Project, Program, Portfolio, Product Life Cycle, Project Life Cycle, Proposals, Investment ,Decisions, Planning Horizon, Price and Pricing, Cost and Costing, Performance Measurement, Earned Value Management, Termination Decisions, Replacement and Retirement Decisions.

Module:3	Algorithmic Models for Software Cost Estimation	7 hours

Putnam SLIM Model, Doty model, RCA Price Model, COCOMO Model, IBM-FSD Model

Module:4	**Risks and Uncertainty**	**6 hours**
Goals, Estimates, and Plans, Estimation Techniques, Addressing Uncertainty, Decisions under Risk, Decisions under Uncertainty		
Module:5	**Economic Analysis Methods**	**6 hours**
For-Profit Decision Analysis, Minimum Acceptable Rate of Return, Return on Investment, Return on Capital Employed, Cost-Benefit Analysis, Cost-Effectiveness Analysis, Break-Even Analysis, Business Case, Multiple Attribute Evaluation, Optimization Analysis		
Module:6	**Software eco system**	**6 hours**
Software ecosystem overview, Supplier relationships in software eco system, associative models and case studies		
Module:7	**Software business case**	**7 hours**
Business case overview, Steps of business case process, Developing business cases, Tying the business process with SDLC, Principles, rules and analysis tools for a making business case		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1. Karl Popp, Advances in Software Economics: A Reader on Business Models and Partnering, Books on Demand, 2011.

Reference Books

1. Guide to Software Engineering Body of Knowledge Version 3.0 – IEEE Computer Society-chapter 12
2. Barry W.Boehm, Software Engineering Economics, IEEE transactions on Software Engineering,
3. Donald J. Reifer ,Making the Software Business Case: Improvement by the Numbers (SEI Series in Software Engineering), Addison Wesley, 2001

Recommended by Board of Studies	5-3-2016		
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE2029	Agile Development Process	L	T	P	J	C
		3	0	0	0	3
Pre-requisite	SWE1701	Syllabus version				
		v. 1.0				

Course Objectives:

1. To methodology and issues
2. To learn the fundamental principles and practices associated with various agile development methods
3. To learn how agile methods scale to large and distributed projects, including the role of systems engineering

Expected Course Outcome:

1. Understand of agile software engineering and its advantages
2. Understand software engineering standards for Agile process
3. To apply agile software engineering practices over the entire software development lifecycle..
4. To compare various Agile Methodologies
5. Understand Scrum Framework and its application scenarios.
6. To understand Agile Metrics Release Planning and Estimation in Scrum based software development.
7. Understand how agile methods scale to large and distributed projects

Module:1	INTRODUCTION TO AGILE	6 hours

Introduction to Agile Software Process Model - Agile Methodology & Principles – Types – Benefits - Life Cycle, Agile Project Management – Design and Construction - Agile Testing- Agile Tools.

Module:2	AGILE PROCESSES	6 hours

Key Process Areas in CMM – Quality Improvement – Six Sigma : Six Sigma Overview, DMAIC - Define, Measure, Analyze, Improve, Control; DMADV -Define, Measure, Analyze, Design, Verify; Lean : Lean Overview, Lean Principles, Lean Rules, Lean Implementation - The 8 Forms of Waste; Lean Tools - 5 Why's, Pareto.

Module:3	AGILE REQUIREMENTS	6 hours

Meeting the requirements challenge iteratively-Requirements for Agile approach – Gathering & analysis –Behavior Driven Development (BDD) and Acceptance Test Driven Development (ATDD)- Designing storyboards and scrums in Agile approach.

Module:4	AGILE METHODOLOGIES	8 hours

Pair Programming – Refactoring – Dynamic Systems Development (DSD) – Feature Driven Development (FDD) – Test Driven Development (TDD), Agile Unified Process – Agile Failure Models - Various reasons why agile fails?

Module:5	SCRUM	7 hours

Scrum Foundations - Scrum Roles - Scrum Master - Product Owner – Team - Scrum Meetings - Scrum Artifacts - Product Backlog - Sprint Backlog - Burn-down Charts - Scaling Scrum – Manager in Scrum and Product Backlog.

Module:6	**AGILE PLANNING and ESTIMATION**	**5 hours**
Principles of Agile Metrics – Release, Planning and Estimation in Scrum.		
Module:7	**ADVANCED CONCEPTS & CASE STUDIES**	**5 hours**
Scrum and Large Projects – Distributed Scrum – Agile Adoption - A case study of a scrum project, Scrum Success Stories.		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)	
1.	K.S. Rubin, Essential Scrum: A Practical Guide to the Most Popular Agile Process, Addison-Wesley, 2012.
Reference Books	
1.	M. Cohn, Succeeding with Agile: Software Development Using Scrum, Addison-Wesley, 2009
2.	S.W. Ambler, M. Lines, Disciplined Agile Delivery: A Practitioner's Guide to Agile Software Delivery in the Enterprise, IBM Press, 2012.
3.	Chetankumar Patel, Muthu Ramachandran, Story Card Maturity Model (SMM): A Process Improvement Framework for Agile Requirements Engineering Practices, Journal of Software,Academy Publishers, Vol 4, No 5 (2009), 422-435, Jul 2009.
4.	Kevin C. Desouza, Agile information systems: conceptualization, construction, and management,Butterworth-Heinemann, 2007
5.	K. Beck, C. Andres, Extreme Programming Explained: Embrace Change, 2nd Edition, Addison-Wesley, 2004.

Recommended by Board of Studies		5-3-2016	
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE2030	REVERSE ENGINEERING	L	T	P	J	C
		3	0	0	0	3
Pre-requisite	SWE1701	Syllabus version				
		v. 1.0				

Course Objectives:

1. To provide a broad introduction to Reverse Engineering and their programming.
2. To explain and apply the fundamental concepts and terminology of Reverse Engineering.
3. To explain and address the fundamental problems of Reverse Engineering.

Expected Course Outcome:

1. Have a clear understanding about reverse engineering concepts
2. Study about the different programming aspects for reverse engineering
3. Attainment of knowledge about various reversing tools
4. Address about protection breaking and cracking
5. Study about disassembly process
6. Understand and apply object oriented approach for reverse engineering
7. Deeper understanding and applications using java programming for reverse engineering
8. Knowledge about industry standard reverse engineering

Module:1	Foundations of Reverse Engineering	4 hours

Reverse Engineering, Software Reverse Engineering, Reversing Applications, Is Reversing is legal

Module:2	Low Level Software and windows fundamentals	7 hours

Reversing process, Low Level Software-High-Level perspectives, Low level perspectives, Assembly language, A primer on compilers and compilation, Execution Environments

Module:3	Reversing Tools	6 hours

Reversing Approaches, Disassemblers, Debuggers, Decompilers, System-Monitoring Tools, Patching Tools, Miscellaneous Reversing Tools

Module:4	Cracking	6 hours

Piracy and copy protection, Antireversing techniques, Breaking protections

Module:5	Disassembly	6 hours

Reversing.Net, Decompiling classes, Obfuscating classes

Module:6	Object oriented code-I	6 hours

Accessing Non-Public methods and variables of a class, Replacing and patching Application classes		
Module:7	**Object oriented code-II**	**8 hours**
Manipulating java security, Reverse engineering applications, Intercepting Control flow, Software Production.		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1. Eldad Eilam Reversing Secrets of Reverse Engineering, Wiley Publishing,Inc, 2011

Reference Books

1. Alexandre Gazet, and Elisas Bachallany ,Practical Reverse Engineering X86, X64,ARM, Windows, Kernel, Reversing Tools and Obfuscation by Bruce Dang, Wiley 2014
2. Paolo Tonella, Alessandra Potrich ,Reverse Engineering of Object Oriented Code by Springer Science 2005
3. Covert Java Techniques for Decompiling,Patching and Reverse Engineering by Alex Kalinovsky, SAMS Publishing 2004

	Recommended by Board of Studies		5-3-2016	
	Approved by Academic Council	No. 40th	Date	18-3-2016

SWE2031	Global Software Engineering	L	T	P	J	C
		3	0	0	0	3
Pre-requisite	**SWE1701**	**Syllabus version**				
		v. 1.0				

Course Objectives:

1. The objective of this course is to provide knowledge, competence and practical experience regarding communication, cooperation and coordination among distributed teams while performing software engineering activities.
2. The student will learn how to communicate on a global network, in a global team, and interpret and sensitively exploit diversity in their professional life.
3. The students will gain the generic skills such as problem solving, decision making, teamwork and understanding of cultural diversity.

Expected Course Outcome:

1. Understand the benefits of offshoring / outsourcing
2. Design a software system and its process to meet user needs
3. Able to identify the appropriate tools and techniques useful for global software engineering.
4. Understand the project management and project co-ordination techniques for global software projects.
5. Understand the challenges involved in global software development.
6. Analyze software risks and identify mitigation strategies.
7. Evaluate processes and products against the applicable standards and metrics.
8. Understand the available advanced process models for enhancing the business.

Module:1	Product Development Strategy	6 hours

Different Business Models, The Bright Side: Benefits, The Dark Side: Challenges, Deciding the Business Model, Preparing the Business Case.

Module:2	Product Planning and Development	6 hours

Requirements Engineering, Establishing the Groundwork, Eliciting Requirements, Building the Requirements Model, Estimation and Planning, Development Processes.

Module:3	Global Software Architecture	5 hours

Global Software Architecture Development, Practice: Software Chunks and Distributed Development, Configuration Management, Open Source Development, Quality Control, Tools and IT Infrastructure, Practice: Collaborative Development Environments

Module:4	Vendor Management	6 hours

Life cycle Management, Supplier selection and Evaluation, Supplier Management, Practice: IT Outsourcing – A supplier perspective, Monitoring Cost, Progress and Performance.

Module:5	**Risk Management**	**6 hours**
Risk Management, Practice: Risk Assessment in Globally distributed projects, Intellectual property and Information security, Practice: Global Software Engineering in Avionics, Practice: Global Software Engineering in Automotive.		
Module:6	**People and Teams**	**6 hours**
Work Organization and Resource Allocation, People involved: Roles and Responsibilities, Soft skills, Practice: People factors in Globally distributed projects, Practice: Requirements Engineering in Global teams, Practice: Educating Global Software Engineering-Digital rights management.		
Module:7	**Advancing Your own Business**	**8 hours**
Key take-away tips, Global software and IT rules of Thumb, The world remains flat, Managing cultural and language differences, Infrastructure support for Global software development. **Agile software development with distributed teams:** Scrum in distributed environments, Agile adoption, Scrum success stories		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1. Christof Ebert, Global Software and IT: A Guide to Distributed Development, Projects, and Outsourcing, 1st Edition, Wiley-IEEE Computer Society, 2011.

Reference Books

1. Erran Carmel, Global software Teams Collaborating across Borders and Time zones, 1st Edition, Pearson Prentice Hall, 1999
2. Raghvinder Sangwan, Matthew Bass, Neel Mullick, Daniel J. Paulish, Juergen Kazmeier, Global Software Development Handbook, 1st Edition, CRC Press, 2006
3. Elizabeth Woodward, SteffanSurdek, Matthew Ganis, A Practical Guide to Distributed Scrum (IBM Press), 1st Edition, Prentice Hall, 2010.

	Recommended by Board of Studies		5-3-2016	
	Approved by Academic Council	No. 40th	Date	18-3-2016

SWE2032	KNOWLEDGE ENGINEERING	L	T	P	J	C
		3	0	0	0	3
Pre-requisite	**SWE1701**	**Syllabus version**				
		v. 1.0				

Course Objectives:

1. To learn the fundamentals of Knowledge Engineering concepts.
2. To represent the real-world concepts in terms of Knowledge Units
3. To design & develop a Knowledgebase for Experts Systems
4. To apply Knowledge Engineering principles across

Expected Course Outcome:

1. Understand the fundamentals of knowledge engineering process
2. Know the different knowledge representation models
3. Design customized representation models for knowledge reasoning
4. Solve problems in reasoning knowledge for modelling expert systems
5. Develop production systems, description logic-based systems and Bayesian networks
6. Use logic in knowledge representation, reasoning and planning
7. Design knowledgebase for expert systems

Module:1	**Basics of Knowledge Processes**	**6 hours**

Knowledge - concepts, relations, Types of Knowledge – Tacit, Explicit, Implicit, Hybrid, Knowledge Processes – acquisition, representation, reasoning, storing, sharing, reuse.

Module:2	**Knowledge Acquisition and Expression**	**6 hours**

Repositories – structured, semi-structured, unstructured. Introduction to knowledge representation and reasoning, role of logic, the language of First orders logic and knowledge-Based systems, Knowledge Engineering and Expressing Knowledge.

Module:3	**Knowledge Representation**	**5 hours**

The propositional case, handling variables and quantifiers, dealing with computational intractability. Clauses, Concepts, Relations, Knowledge Units, Representation.

Module:4	**Procedural Control of Reasoning and Rules**	**6 hours**

Horn Clauses, SLD resolution, Computing SLD derivations. Facts and rules, Rule formation and search strategy, algorithm design, specifying goal order, committing to proof methods, controlling backtracking, negation as failure, Dynamic databases.

Module:5	**Production Systems & Representation**	**7 hours**

Production systems, working memory, production rules, conflict resolution, making production systems more efficient. Objects and frames, a basic frame formalism, an example: Using frames to

plan a trip, beyond the basics. Case study: Dr. Watson, Deep Blue		
Module:6	**Structured Descriptions, Inheritance and Defaults**	**6 hours**
Descriptions, Meaning and entailment, Computing entailments, taxonomies and classification, Inheritance network, strategies for defensible inheritance. Introduction to defaults, closed-world reasoning, circumscription, default logic, Autoepistemic logic.		
Module:7	**Design of Knowledgebase**	**7 hours**
Knowledgebase Architecture, The layered approach to design KB, Logical Entailment, Conceptual Graph for KB – constructions, updation, deletion, traversal. Case study- Expert Systems Design with KB.		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1. Grega Jakus, Veljko Milutinovic, Sanida Omerovic, Saso Tomazic, ‒Concepts, Ontologies, and Knowledge Representation‖, Springer, 2013
2. Ronald J. Brachman and Hector J.Levesque, ‒Knowledge representation and reasoning‖, 2nd edition, Elsevier publications, 2004.

Reference Books

1. Ngoc Thanh Nguyen,‖Advanced Methods for Inconsistent Knowledge management‖ Springer, ISBN-13: 978-1849966672, 2010.
2. Simon Kendal, Malcolm Creen, ‒An Introduction to Knowledge Engineering, Springer, ISBN-13: 978-1846284755, 2007
3. Schneider Kurt, ‒Experience and Knowledge Management in Software Engineering, Springer, ISBN 978-3-540-95880-2, 2009
4. Ulla de Stricker, ‒Knowledge Management Practice in Organizations: The View from Inside‖, de Stricker Associates Canada, 2014

	Recommended by Board of Studies		5-3-2016	
	Approved by Academic Council	No. 40th	Date	18-3-2016

SWE2034	Ruby Programming	L	T	P	J	C
		3	0	2	0	4
Pre-requisite	CSE1002	Syllabus version				
		v. 1.0				

Course Objectives:

1. Understand the syntax and semantics of the Ruby language and their similarity and differences from Java.
2. Understand how to develop and implement various types of programs in the Ruby language.
3. Understand various forms of data representation and structures supported by the Ruby language.
4. Understand the appropriate applications of the Ruby language.

Expected Course Outcome:

1. Understand the basic fundamentals and structure of Ruby
2. Object Oriented approaches and Interfaces
3. Understanding and implementing the storage structures of Ruby
4. Handling data using Files to process and store data
5. Ability to build, manage and schedule multiple processes
6. Testing and solving various exception errors in a module
7. Integrating Remote System connectivity using Socket Programming

Module:1	Getting Started with Ruby	4 hours

Introduction – Structure and Execution of Ruby Programming – Data types and Objects – Expressions and Operations – Statements and Control Statements

Module:2	Classes, Objects and Methods	8 hours

Classes and Objects - Methods – Procs – Lambdas and Closures – Modules, Namespaces and Mix-Ins – Blocks and Iterations - Reflection and Meta Programming

Module:3	Ruby's Building Blocks	6 hours

Arrays – Collection handling with Arrays – Hashes – Ranges - String - Numbers - Math - Container

Module:4	Files and Directories	5 hours

Input and Output Objects - Files and Directories – Opening and Closing of Files – Reading and Writing Files

Module:5	Fibers, Threads and Processes	6 hours

Fibers – Multi –Threading – Scheduling – Mutual Exclusion – Running multiple Processes–

Database		
Module:6	**Exceptions and Testing**	**7 hours**
Documentation – Exceptions, Catch and Throw – Handling Exception – Raising Exception - Testing : Unit Testing - Assertions – Bench Marking and Profiling		
Module:7	**Networking and Sockets**	**7 hours**
Networking – Network Operations – Simple TCP Server – Multi-Client TCP Server – Daemon Processes		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1. Programming Ruby 1.9 and 2.0- The Pragmatic Programmers' Guide (Facets of Ruby) , 4th Edition, Dave Thomas, with Chad Fowler and Andy Hunt , 2013.
2. Beginning Ruby: From Novice to Professional (Expert's Voice in Open Source)‖, 3rd edition, Peter Cooper, 2016

Reference Books

1. The Well-Grounded Rubyist: Covers Ruby 1.9.1‖, 1st Edition, David A. Black, 2009
2. Eloquent Ruby (Addison-Wesley Professional Ruby)‖, 1st Edition, Russ Olsen,2011.

List of Challenging Experiments (Indicative)

1. **Arrays and Hashes**

   Create a program that gives a personalized greeting. There should not be any truly ―interactive‖ elements to the program itself, so the information in the greeting will have to be static. The method should greet a person as such:

   ―Why hello there Bob, my name is Sue.‖

   The first underlined element should be the value of the input argument for your method, while the second should be your global variable value.

   The goal is to utilize 1 method call and 1 global variable. The global variable should not be maintained inside of the method. Also, try to utilize any shortcuts you may have found during the reading. Comment accordingly.

2. **Classes and Objects**

   You will need to keep track of the name, cost, vending number, and supply of each vending food object. The child classes should be utilizing their parent‘s initialization method. Do not ask for the supply count when creating a new object, this will be done later via method calls. Add in attribute readers and writers for the

	instantiated variables for testing purposes. Now you need to give Vendor Food the ability to stock an item, which should accept the number of items desired to be added and update the supply accordingly. Next, add in the ability to vend an item via another method call, which depletes the supply by 1 if the supply exists to do so (do nothing otherwise, error handling is not needed yet). Since we are selling items now we should keep track of how much we make, create a sales updater in the Vendor Food class which increments the cost of the vended item to a class variable designated to storing the total sales value. Additionally, create a class method to view the sales information. Lastly, redefine the ―to string‖ method in all classes. In the parent class give the basic attribute information, but in the child classes, make sure to call the parent‘s to string and tack on some text identifying which class the ―to string‖ called from.	
3.	**Containers, Blocks and Iterations** Decided that our current sales log setup is no longer sufficient as it only tells us how much we‘ve sold and with no regard to what was sold. Also, we would like a new way to input our added inventory so we‘ll update that functionality as well. Since our sales are not itemized, we want to keep an active working array of the items sold. Create a new class variable to contain this array. In the sales updating function add the current Vendor Food object to the sales array. Now that we have an array containing our sold objects (in the order they occurred) we should add a function to save the sales array to a sales log file, for backup purposes. Simply iterate through the array (utilizing block calls) and save the object data (name, cost, vent number) to file in a delimited format. (The ― | ― symbol is an excellent choice). *You will need to use File.open(filename, ―r+‖) and you will need to create an empty log file in the working directory, until we learn more about files. Also, filename.puts ―‖ will write to your file. To aid in the readability of this log file, create 2 log reader functions in the Vendor Food class. The first should output a cleaned up version of the sales log. The second function should receive a snack name or vending number and return the number of times items matching that criteria were sold.	
4.	**Regular Expressions and Methods** Implement the famous concept, the game of Hangman. The game will be a standalone application driven by user input. The game is only required to run through once per execution Exceptions Handling	

	Input / Output Functions Unit Testing and Debugging	
5.	The objective is to create 4 classes, in separate files, representing the Circle, Square, Triangle, and Point. Each of the shape classes will require the Point class. The point is simply used to store/retrieve the 1-4 points associated with each shape (an X and Y coordinate in each). Each of the shape classes is to define a points array and area as instance variables. Each shape class will define a unique initialize method. For the Circle, require the x,y cords of the center point and a radius. The Square will require the x, y coords of the lower-left point, a width, and a height. The Triangle will require the x,y cords of the lower-left point, a base, and a height. For the Square and Triangle, calculate the values of the remaining points from the supplied data, and fill the points array of the corresponding class with these Point objects (the Circle will have a points array with one Point object). Next, create a separate file to house the Shapes module. This module will define some common functions, which you may want to use with shapes.	
	Total Laboratory Hours	30 hours

Recommended by Board of Studies		12-8-2017	
Approved by Academic Council	No. 47th	Date	5-10-2017

SWE2035	Big Data Technologies	L	T	P	J	C
		3	0	2	0	4
Pre-requisite	**SWE1004**	**Syllabus version**				
		v. 1.0				

Course Objectives:

1. To understand the basics of big data analytics concepts
2. To explore tools and practices for working with big data

Expected Course Outcome:

1. To learn about Big data, its characteristics and analytics life cycle
2. To understand the challenges in storing big data and how it is resolved
3. To understand the limitation of systems in processing big data and how it is overcome
4. To develop Map Reduce Programs
5. To learn about tools in Ecosystem for analysing big data
6. To practice Hive queries and write scripts to analyse big data
7. To apply the big data technologies for solving real world problems

Module:1	Introduction to Big Data	5 hours

Big Data Overview – Characteristics of Big Data –Business Intelligence v/s Data Analytics – Need of Data Analytics – Data Analytics in Industries – Role of the Data Scientist – Data Analytics Life Cycle. Evolution of Big data – Best Practices for Big data Analytics - Big data characteristics - Volume, Veracity, Velocity, Variety

Module:2	Introduction to Hadoop & HDFS	7hours

Overview of Hadoop – Need of Hadoop – Hadoop Eco System - The Distributed File System: HDFS, – The Design of HDFS – HDFS Concepts – Working with HDFS

Module:3	Hadoop Architecture	9hours

Hadoop Deamons - Hadoop Cluster Architecture – HDFS Data Flow– Working of MapReduce — Map and Reduce Phase – Job Processing in Hadoop

Module:4	Map Reduce Programming	5hours

Developing MapReduce Program – Block vs Split Size – Input output format – Key, Text, Sequence, NLine file format, XML file format

Module:5	Map Reduce Features	7hours

Counters – Sorting – Partial sort – Total sort - Secondary Sorting – Map side join and Reduce side join – Side data distribution : distributed cache and configuration

Module:6	**Hadoop EcoSystem**	**5hours**
Apache Hive Fundamentals Introduction-Hive modules, Data types and file formats, Hive QL-Data Definition and Data Manipulation		
Module:7	**Querying with Hive**	**5hours**
Hive QL queries, Hive scripts. Aggregate functions. Bucketing vs Partitioning.		
Module:8	**Contemporary issues** .	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1. Tom White, "Hadoop: The Definitive Guide", Third Edition, O'Reilley, 2012.
2. Jason Rutherglen, Dean Wampler,Edward Caprialo, ―Programming Hive‖, O'Reilly Media Inc, 2012.

Reference Books

1. Vignesh Prajapati, Big data analytics with R and Hadoop, SPD 2013
2. Eric Sammer, "Hadoop Operations", O'Reilley, 2012.

List of Challenging Experiments (Indicative)

1. Setting up Hadoop in Single node / Multinode Environment
2. Working with HDFS using Commands
3. Simple Program using MapReduce
4. MapReduce Program to show the need of Combiner
5. Custom Partitioning
6. MapReduce I/O Formats –Text, key- value
7. MapReduce I/O Formats – Nline
8. Sequence file Input / Output Formats
9. Top K records
10. Side data by configuration
11. Map side join and Distributed Cache
12. Reduce side Join
13. Program using Hive manipulation and data definition languages.
14. Program using Hive queries with partitioning.

		Total Laboratory Hours	30 hours
Recommended by Board of Studies		12-8-2017	
Approved by Academic Council	No. 47th	Date	5-10-2017

Course code	Course title	L	T	P	J	C
SWE3003	Sensor Networks	3	0	0	0	3
Pre-requisite	SWE2002	Syllabus version				
		v. 1.0				

Course Objectives:

- To understand the needs of Wireless Sensor Network in current scenario of technology.
- To explain the principles and characteristics of wireless sensor networks.
- To describe current technology trends for the implementation and deployment of wireless sensor networks.
- To discuss the challenges in designing MAC, routing and transport protocols for wireless sensor networks.
- To understand the tools and operating system for wireless sensor networks.

Expected Course Outcome:

Upon Completion of the course, the students will be able to

- Understand the basic knowledge about wireless sensor networks.
- Design/Architect sensor networks for various applications.
- Analyze various communication models for an energy efficient sensor network.
- Decide an appropriate sensor network topology for the effective network deployment.
- Identify suitable routing protocols for wireless sensor networks.
- Compare various transport layer and congestion control protocols and identify a suitable transport layer protocol for real time applications.
- Solve the problems related to the wireless sensor networks and evaluate the performance of sensor networks and identify bottlenecks.

Module:1	Sensor technology fundamentals	5 hours

Sensor, Sensor Characteristics, Sensor Node Architecture, Sensor Network Architecture, Mote Technology, Compare MANET and WSN, Requirement of WSN.

Module:2	Overview of Wireless Sensor Networks	5 hours

Challenges for Wireless Sensor Networks-Characteristics requirements-required mechanisms, Difference between mobile ad-hoc and sensor networks, Applications of sensor networks-Enabling Technologies for Wireless Sensor Network.

Module:3	Wireless Sensor Network Architecture	6 hours

Single-Node Architecture - Hardware Components, Energy Consumption of Sensor Nodes, Operating Systems and Execution Environments, Network Architecture - Sensor Network Scenarios, Optimization Goals and Figures of Merit, Design Principles of WSN, Gateway Concepts.

Module:4	Communication Protocols	6 hours

Physical Layer and Transceiver Design Considerations, MAC Protocols for Wireless Sensor Networks, Low Duty Cycle Protocols, Contentation and Schedule based Protocols, Link-Layer Protocol, Address and Name Management, Assignment of MAC Addresses.

Module:5	WSN Infrastructure Establishment	6 hours

Time Synchronization, Localization and Positioning, Topology Control, Transport layer and QoS in WSN, Reliable Data Transport, Congestion and rate control.

Module:6	**WSN Routing Protocols**	**7 hours**
Faces of Forwarding and Routing Protocols, Energy-efficient Unicast, Broadcast and Multicast, Geography Routing, Mobile Nodes, Data-centric routing, Data-centric storage.		
Module:7	**Sensor Network Application, Tools and OS**	**7 hours**
WSN Advanced Application Support - Advanced in-network processing, Security, Application-specific support, WSN OS Introduction - Examples of Operating Systems: Tiny OS, Mate, Magnet OS		
Module:8	**Contemporary issues**	**3 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1.	Holger Karl and Andreas Wiilig, ―Protocols and Architectures for Wireless Sensor Networks – Student Edition‖ John Wiley & Sons Limited 2012.

Reference Books

1. Jacob Fraden ―Handbook of Modern Sensors‖, Fourth Edition, Springer Publiser – 2010.
2. Mukherjee N, Neogy S, Roy S. ―Building Wireless Sensor Networks: Theoretical and Practical Perspectives‖ - CRC Press Book – 2015
3. Akyildiz IF, Vuran MC. ―Wireless Sensor Networks‖. Wiley; 1 edition. Published 2010.
4. Carlos de Morais Cordeiro and Dharma Prakash Agrawal, ―Ad Hoc and Sensor Networks: Theory and Applications‖, Second Edition, World Scientific Publishers, 2011
5. Dargie WW, Poellabauer C. Fundamentals of Wireless Sensor Networks: Theory and Practice. Wiley Publication – 2010.

	Recommended by Board of Studies		5-3-2016	
	Approved by Academic Council	No. 40th	Date	18-3-2016

SWE 3005	SOFTWARE QUALITY AND RELIABILITY	L	T	P	J	C
		3	0	0	0	3
Pre-requisite	**SWE2005**	**Syllabus version**				
		v.1.0				

Course Objectives:

1. To introduce the importance of Quality of Software Products
2. To elicit, analyze, prioritize, and manage both functional and quality requirements
3. To plan for Software quality assurance
4. To learn the concepts of Reliability
5. To understand and apply configuration and quality management techniques in software development processes

Expected Course Outcome:

1. To understand the significance of software quality assurance in software projects.
2. To understand and know how to manage software quality in software organizations.
3. To understand and apply software quality assurance metrics in software projects.
4. To implement software quality programs in software projects
5. To understand and apply software standardization in software projects.
6. To apply and practice software reliability techniques.
7. To understand software reliability engineering process
8. To identify contemporary issues in applying software quality and reliability techniques.

Module:1	**Fundamentals of Software quality Assurance**	**7 hours**

The Role of SQA- Software Quality Assurance Plan-Software Quality Assurance considerations-Need of Software Quality Assurance -SQA People

Module:2	**Managing Software Quality**	**7 hours**

Quality Management-Software Configuration Management-Managing Software organizations-Managing Software quality –Defect Prevention

Module:3	**SQA Metrics**	**6 hours**

Software Quality-Total Quality Management (TQM)-Quality Metrics-Software Quality metric Analysis

Module:4	**Software Quality Program**	**5 hours**

Software quality program Concepts-Establishment of a software quality program-Software Quality Assurance planning-purpose and scope of Software Quality Program

Module:5	**SQA Standardization**	**6 hours**
Software standards-ISO 9000 Quality system standards-Capability Maturity model and the Role of SQA in software development maturity- Six Sigma Concepts		
Module:6	**Reliability Concepts**	**5 hours**
Reliability Definition-Quality and Reliability-Reliability Functions-Reliability Mathematics-Measures of Reliability		
Module:7	**The Reliability Engineering Process**	**7 hours**
Defining the product-Testing the acquired software-Learning reliability concepts-s/w and h/w reliability		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1. Mordechai Ben-Menachem / Garry S Marliss, ―Software Quality‖, Vikas Publishing House, Pvt, Ltd., New Delhi,2014.

Reference Books

1. Meir Liraz,‖Quality Assurance:How to set up and manage a Quality Control System‖,Kindle Edition,2013
2. Solis Tech,‖ Quality Assurance:Software Quality Assurance made easy‖,Kindle Edition,2016
3. Watts S Humphrey, ― Managing the Software Process‖, Pearson Education Inc,2007
4. John D Musa, ―Software Reliability Engineering‖,1998
5. Gordon G Schulmeyer, ―Handbook of Software Quality Assurance‖, Third Edition, Artech House Publishers, 2007.
6. Charles E. Ebeling, ―An introduction to Reliability and Maintainability engineering‖, TMH, 2000.
7. RoyBillington and Ronald N. Allan, ―Reliability Evaluation of Engineering Systems‖, Springer, 2007.

Recommended by Board of Studies	5-3-2016		
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE3006	ADVANCED SOFTWARE TESTING	L	T	P	J	C
		3	0	2	0	4
Pre-requisite	**SWE2005**	**Syllabus version**				
		v.1.0				

Course Objectives:

1. To know the concepts of testing in SDLC.
2. To understand testing practices in industry related to functional and non-functional domains.
3. To have an exposure to specialized testing tools and techniques

Expected Course Outcome:

1. Ability to apply software testing techniques in process of SDLC and engineering methods.
2. Examine and solve various functionality problems by designing and selecting testing models and methods in software project management
3. Examine and solve various program logic or structure problems, by designing and selecting testing models and methods - functional testing
4. Examine and solve various program logic or structure problems, by designing and selecting testing models and methods - Nonfunctional testing
5. Developing and testing the applications with various automation tools both functional and nonfunctional testing - test automation
6. Apply the knowledge on testing and creating test reports based on the automation tools
7. Develop construct the complementary techniques to dynamic testing for improving the software quality

Module:1	BASIC CONCEPTS IN SOFTWARE TESTING	7 hours

Overview of Testing Techniques - Types of Software Testing – Role of Testing in SDLC, Testing Life Cycle (TLC), Testing Strategies and Tactics, Creating Test Plans and Test Cases – Test scenarios – Test Data – Test Scripts, Test Requirements Specification – Requirements gathering – Creating TRS and Test Procedure

Module:2	LIFE CYCLE TESTING & TEST PROJECT MANAGEMENT	7 hours

SDLC Testing – Testing in the Requirement Phase - Logical & Physical Design Phase, Test Project Management – Estimating Test Costs and Duration – Staffing - Testing Team, Building a Software Testing Environment – Creating an environment supportive of software testing – Building Software Testing Process – Selecting and Installing Software Testing Tools – Building Software Tester Competency

Module:3	SOFTWARE FUNCTIONAL SYSTEM TESTING	5 hours

Functional Testing – Automated Unit Testing – Test Plan & Scripts – White Box Testing – Black Box Testing – Creating Automated Test Procedures and Reports – Integration Testing – Order of Integration – OO System Integration – Creating & Maintaining Tested Databases

Module:4	**SOFTWARE NON-FUNCTIONAL SYSTEM TESTING**	**5 hours**
Non-Functional Testing – Performance Testing – Load Testing – Endurance Testing – Scalability Testing – Volume Testing - Security Testing – Internationalization Testing – Creating Test Procedures and Reports – Test Plans – Creation of Data-pool, Bottleneck Identification – Performance Analysis and Reporting		
Module:5	**TOOLS AND ITS APPLICATION IN SPECIFIC TESTINGS**	**6 hours**
Automated Testing Tools – Functional Testing - Rational Functional Tester – Selenium – Cucumber - JUnit, Performance Testing Tools - Rational Performance Tester – HP Load Runner, Test Management Tools - Quality Center, Performance Center		
Module:6	**REPORTS AND REVIEWS**	**6 hours**
Reports and Control Issues – Types of Review – Component of Review Plans – Reporting Review Results – Evaluation of Software Quality		
Module:7	**ADVANCED CONCEPTS IN SOFTWARE TESTING**	**7 hours**
Test Coverage and Test Metrics Management, Improving the Test Processes – Test Process Optimization, Empirical Software Testing and Analysis, SOA Testing – General Principles and Procedures, Data Warehouse Testing, Cloud Testing, Big Data Testing, Web Apps Testing		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1.	Glenford J. Myers, Corey Sandler, Tom Badgett - The Art of Software Testing, 3rd Edition, 2011

Reference Books

1.	Aditya P. Mathur , –Foundations of Software Testing: Fundamental Algorithms and Techniques‖, Pearson Education India, 2007
2.	Doug Vucevic & Wayne Yaddow, –Testing the Data Warehouse Practicum: Assuring Data Content, Data Structures‖, Trafford Publishing, 2012
3.	Scott Tilley , Tauhida Parveen, –Software Testing in the Cloud: Migration and Execution‖, Springer, 2012
4.	Nageshwar Rao Pusuluri, –Software Testing Concepts and Tools‖, DreamTech Press, Reprint Edition 2008.
5.	Anne Mette Jonassen Hass, ―Guide to Advanced Software Testing‖, Artech House, 2008.
6.	William E. Perry , –Effective Methods for Software Testing: Includes Complete Guidelines‖ 3rd Edition, Wiley Publications, 2006
7.	William E. Lewis , –Software Testing and Continuous Quality Improvement‖, Third Edition, Auerbach Publications, 2008

List of Challenging Experiments (Indicative)		
1.	Write the Procedure for RPT. Record the test for VIT intranet portal with some 10 links and Create Performance Schedule and generate the Test Report for the same.	
2.	Design a selenium web driver program to handle pop ups. Go to student login page, click on login button without giving username and password, and handle that pop up message	
3.	Imagine a program which reads in the length of three sides of a triangle and outputs a message naming the kind of triangle: EQUILATERAL, ISOCELES or SCALENE. Length not in range 1 - 99 cause error message INVALID INPUT. If lengths don't make a triangle, output NOT A TRIANGLE. Assumptions (pre-conditions for the program) Three lengths are entered separated by blanks or returns. Input of decimals or characters causes unpredictable results. Input from keyboard, simple text output to display. Even though equilateral triangle is also isosceles, only print EQUILATERAL. Write the Junit Test cases for above given logic.	
	Total Laboratory Hours	30 hours

Recommended by Board of Studies		5-3-2016	
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE 4001	System Programming	L	T	P	J	C
		3	**0**	2	0	**4**
Pre-requisite	**SWE 3001**	**Syllabus version**				
		v. 1.0				

Course Objectives:

1. To understand the relationship between system software and machine architecture.
2. To study the architecture of a hypothetical machine, its assembly language, macro language
3. To know the design and implementation of assemblers.
4. To know the design and implementation of Linkers and Loaders.
5. To understand macro processors

Expected Course Outcome:

1. Program in assembly language
2. Implement a symbol table with functions to create, insert, modify, search, and display
3. Have an understanding of foundation to design of SIC & SIC/XE Machine Architecture
4. Have an understanding of foundation to design of assemblers
5. Implement the understood design of macro processors loaders and linkers concepts as programs
6. Examine what happens during program compilation, linking, and loading using C Programming
7. Understand the concepts and theory behind the implementation of high level programming languages
8. Understand the concepts and theory behind the implementation YACC compiler programming

Module:1	**An Overview of System Programming**	**6 hours**

System software and System programming- Views of System Software, and Programming Languages and Language Processors.

Module:2	**Machine Architectures**	**6 hours**

Programming systems, Simplified Instructional Computers (SIC) – SIC Machine Architecture, SIC/XE Machine Architecture, SIC Programming Examples; Traditional Complex Instruction Set Computer (CISC) Machines – VAX Architecture, Pentium Pro Architecture; RISC Machines – Ultra SPARC Architecture, PowerPC Architecture, Cray T3E Architecture.

Module:3	**Assemblers**	**6 hours**

A Simple SIC Assembler, Algorithm & Data Structures; Machine-dependent Assembler Features

– Literals, Symbol-Definition statements, Expression, Program Blocks, Control Sections and Programming Linking; Assembler Design Options – One-pass assembler programming, Multi-Lpass Assemblers programming-Programming using MASM.		
Module:4	**Loaders and Linkers**	**6 hours**
Basic Loader Functions – Design of an Absolute Loader, A Simple Bootstrap Loader; Boot strap Loader programming, Absolute Loader programming; relocating Loader programming, Machine -Dependent Loader Features – Relocation, Program Linking, Algorithm and Data Structures, Linkage Loader; Machine-independent Loader Features – Automatic Library Search, Loader option; Loader Design Options – Linkage Editor, Dynamic Linkage.		
Module:5	**Macro Processor**	**6 hours**
Macro- Definition, Expansion, Functions- Algorithm & Data Structures; Machine independent Macro Processor Features –Concatenation of Macro Parameters, Generation of Unique Labels, Conditional Macro Expansion, Keyword Macro Parameters; Macro Processor Design Options – Recursive Macro Expansion, Language Translators.		
Module:6	**Compilers**	**6 hours**
Phases of compiler- Machine Dependent Compiler Features – Intermediate Form of the program, Machine-Dependent Code Optimization; Machine-Independent Code generation and Optimization; Structured Variables, Storage Allocation-Implementation Using LEx and YACC compiler programming		
Module:7	**Editors and Debugging system**	**7 hours**
Text Editors – Overview of Editing Process, User Interface, Editor Structure; Interactive debugging Systems – Debugging functions and Capabilities, Relationship with other parts of the system, User Interface Criteria.		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**45 hours**
Text Book(s)		
1.	Leland L Beck - System Software - An introduction to System Programming" Addison-Wesley -Pearson education Third Edition- 2013.	
Reference Books		
1.Srimanta Pal, ―Systems Programming" , Oxford University Press, 2011. 2.Alfred V Aho, Ravi Sethi, and Jeffrey D Ullman, Compilers : Principles, Technique Tools, Addition Wesley, Pearson Education 2014. 3. R.K. Maurya, G.M.Magar "System Programming", Dreamtech Press, 2015. 4.D M Dhamdhere, System Programming , Tata McGaw Hill Education, 2nd Ed , 2011 5.V. Raghavan, ―Principles of Compiler Design‖, Tata McGrawHill Education Publishers, 2010.		

List of Challenging Experiments (Indicative)		
1.	Implement a symbol table with functions to create, insert, modify, search, and display.	
2.	Implement pass one of a two pass assembler.	
3.	Implement pass two of a two pass assembler.	
4.	Implement a single pass assembler.	
5.	Implement a two pass macro processor	
6.	Implement a single pass macro processor.	
7.	Implement an absolute loader.	
8.	Implement a relocating loader.	
9.	Implement pass one of a direct-linking loader.	
10.	Implement pass two of a direct-linking loader.	
11.	Implement a simple text editor with features like insertion / deletion of a character, word, and sentence.	
12.	Implement a symbol table with suitable hashing	
	Total Laboratory Hours	30 hours

Recommended by Board of Studies		12-8-2017	
Approved by Academic Council	No. 47th	Date	5-10-2017

SWE4002	Cloud Computing	L	T	P	J	C
		2	0	0	4	3
Pre-requisite	**SWE3001**	**Syllabus version**				
		v. 1.0				

Course Objectives:

1. To understand cloud services and deployment models
2. To use virtualization tools and mechanisms
3. To build private cloud environment.

Expected Outcome:

1. Understand cloud services and cloud deployment models
2. Use to test techniques and skills for cloud services
3. Propose suitable virtualization concept, cloud resource management and automation strategies
4. Build and experiment with global exchange of cloud resources
5. Make use of cloud storage systems and develop cloud applications
6. Design and evaluate cloud-based system process and component to meet desired cloud environment
7. Formulate the Policies for cloud security services
8. Summarize the adoption of Cloud environment in a given sector industry

Module:1	**Overview of Computing Paradigm**	**6 hours**

Recent trends in Computing- Grid Computing, Cluster Computing, Distributed Computing, Utility Computing, Web services, Introduction to Cloud Computing- NIST Cloud Computing Reference Architecture.

Module:2	**Cloud Models**	**5 hours**

Characteristics – Cloud Services – Cloud models (IaaS, PaaS, SaaS) – Public vs Private Cloud – Community, Hybrid Clouds

Module:3	**Basics of Virtualization**	**5 hours**

Types of Virtualization - Implementation Levels of Virtualization - Virtualization Structures - Tools and Mechanisms - Virtualization of CPU, Memory, I/O Devices – Virtual Clusters and Resource Management – Virtualization for Data-center automation.

Module:4	**Cloud Environments**	**4 hours**

Google App Engine, Amazon AWS, Azure - Open Source tools. **Cloud Infrastructure-** Architectural Design of Compute and Storage Clouds – Layered Cloud Architecture Development – Design Challenges - Inter Cloud Resource Management – Resource Provisioning and Platform Deployment – Global exchange of cloud resources.

Module:5	**Security Overview**	**8 hours**
Cloud Security Challenges and Risks – Software-as-a-Service Security – Security Governance - Risk Management. **Security Monitoring**-Security Architecture Design – Data Security – Application Security - Virtual Machine Security - Identity Management and Access Control – Autonomic Security		
Module:6	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**30 hours**

Text Book(s)

1. Anthony T Velte, Toby J. Velte, Robert Elsenpeter, " Cloud computing APractical Approach", Tata McGrawHill Publication, First Edition, 2009.

Reference Books

1. Tim Mather, Subra Kumaraswamy, Shahed Latif, " Cloud Security and Privacy – An Enterprise Perspective on Risks and Compliance", O'Reilly Publications, First Edition, 2009.
2. Akex Amies, Harm Sluiman, Qiang Guo Tang, Guo Ning Liu, –Developing and Hosting Applications on the Cloud‖, IBM Press, 2012.
3. Judith Hurwitz , Bloor Robin, Marcia Kaufman & Fern Halper, –Cloud Computing for Dummies‖, Wiley Publications, 2009.
4. George Reese, –Cloud Application Architectures: Building Applications and Infrastructure in the cloud‖, O'Reilly.

Recommended by Board of Studies			5-3-2016
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE4003	Distributed Computing	L	T	P	J	C
		3	0	0	0	3
Pre-requisite	**SWE3001**	**Syllabus version**				
		v.1.0				

Course Objectives:

1. To explore various features of Distributed Computing and its applications in real world scenario
2. To impart knowledge about Remote communication Paradigms in heterogeneous environment
3. To provide an exposure to real world distributed systems.

Expected Course Outcome:

1. Know about the system models and communication between the system
2. Know about the distributed objects and protocols
3. Recognize the inherent difficulties that arise due to distributed environment of computing resources
4. Understanding file services, co-ordination of the system
5. Design a component or a product applying all the relevant standards and with realistic constraints
6. Able to be familiar with the concurrency, security issues of distributed system
7. Understanding the shared memory and distributed operating system
8. Acquire a clear understanding of the subject related concepts and of contemporary issues

Module:1	Introduction to Distributed Systems	5 hours

Introduction to Distributed Systems – Examples of distributed systems, Trends in distributed systems and Challenges. System Models-Physical model, Fundamental model and architecture model

Module:2	Inter process Communications	6 hours

The API for internet protocols, external data representation and marshalling, multicast communication ,issues in the design of IPC

Module:3	Distributed Objects	7 hours

Remote Invocation – Request Reply protocols, Remote procedure call, Remote method Invocation, Java RMI-case study.

Module:4	File system and Services	6 hours

Distributed File Systems –File Service Architecture –Case Study-SUN NFS Distributed File Systems –File Service Architecture –Case Study-SUN NFS Name services – Name services and Domain Name Systems ,Directory Services

Module:5	Coordination and Agreement	6 hours

Distributed Mutual Exclusion algorithms and Election Algorithms. Time and Global states-clocks, events and process states, synchronizing physical clocks, logical time and logical clocks, global state

Module:6	**Transaction and Concurrency Control**	**7 hours**
Transaction , Nested Transactions, Locks, Concurrency Control Distributed Transactions, Atomic commit protocols		
Module:7	**Distributed OS and Shared Memory**	**6 hours**
Distributed Operating System Support -Distributed Shared Memory		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1.	G. Coulouris, J. Dollimore, and T. Kindberg, "Distributed Systems:Concepts and Designs‖, Fifth Edition, Addison Wesley,2012

Reference Books

1. Randy Chow and Theodore Johnson, –Distributed Operating Systems and Algorithms. Addison-Wesley, 2009
2. Mukesh Singhal and N. G. Shivaratri, Advanced Concepts in Operating Systems, Distributed, Database, and Multiprocessor Operating Systems, McGraw Hill, 2008.
3. Pradeep K. Sinha, "Distributed Operating Systems: Concepts & Design", PHI, 2008
4. Andrew.S.Tanenbaum, Maarten Van Steen, ― Distributed Systems –Principles and Paradigms‖, 3e,Second Edition,Prentice Hall -2006

Recommended by Board of Studies	5-3-2016		
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE4004	Geographic information system	L	T	P	J	C
		2	0	0	4	3
Pre-requisite	**SWE3002**	**Syllabus version**				
		v. 1.0				

Course Objectives:

1. Demonstrate an understanding of guidelines, principles, and theories influencing Geographic Information System.
2. To know about the GIS automation and decision making using GIS
3. Use the information sources available, and be aware of the methodologies and technologies supporting the advances in GIS.

Expected Course Outcome:

1. Have a clear understanding of the subject related concepts and of contemporary issues
2. Gain knowledge in Map projections
3. Understand spatial data models
4. Understand the data input errors
5. Have design thinking capability
6. Understand analytical modelling in GIS
7. Use techniques, skills to develop new GIS application

Module:1	**Introduction**	**3 hours**

Overview of Geographic Information Systems:- Definition of a GIS, features and functions; why GIS is important; how GIS is applied; GIS as an Information System; Components of GIS.

Module:2	**Maps**	**5 hours**

Map Projections and Coordinate Systems:-Characteristics of Maps: Map Scale – Classification of Maps; Plane and Geographic Coordinates: Plane Rectangular Coordinate System - Plane Polar Coordinate System – Geographic Coordinate System of Earth; Map Projections: Types of Map Projections – Common Map Projections -Properties - Major uses; Map Projections: Classification -Aspects – Viewpoints; Georeferencing framework – Geodetic and Vertical Datums; Relationship between coordinate system and Map Projections.

Module:3	**Cartography and Spatial data modeling**	**4 hours**

Cartography:- GIS and cartography - Difference between CAD and GIS - Introduction to Remote Sensing-Spatial Data Modelling: Introduction – Entity Definition – Spatial Data Models – Spatial Data Structures: Raster data structures – vector data structure

Module:4	**Data Input and Editing**	**3 hours**

Methods of data input: keyboard entry-manual digitizing-automatic digitizing - Electronic data

transfer - Data editing: Detecting and correcting errors – common errors in spatial data – Re-projection, transformation and generalization – Geocoding address data – Updating and maintaining spatial database [Case study: Ordnance Survey(OS) data collection] - satellite imagery- satellite image resolution and scaling.		
Module:5	**Data analysis**	**5 hours**
Measurements in GIS – lengths, perimeters and areas – Queries – Reclassification - Buffering and neighborhood functions- Integrating data – map overlay - Spatial interpolation- Analysis of surfaces - Network analysis.		
Module:6	**Analytical modelling in GIS**	**4 hours**
Introduction- Process models- Modelling physical and environmental processes - Modelling human processes- Modelling the decision-making process-Problems with using GIS to model spatial processes.		
Module:7	**Data Automation, Database and Data Quality**	**4 hours**
Data Sources: Internet resources for GIS - Data Resources - Product Information - locating and evaluating data - data formats – ArcGIS software; Database- PostGIS database / ArcGIS supported Databases(Ex. PostgreSQL); Data Quality Issues – Introduction, Describing data quality and errors – Sources of error in GIS – Finding and modeling errors in GIS – Managing GIS error.		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**30 hours**
Text Book(s)		
1.	Ian Heywood, Introduction to Geographical Information Systems, Pearson Education, fourth edition, 2012	
2.	C.P.LO, Albert K. W. Yeung, Concepts and Techniques of Geographic Information Systems, Publisher: PHI, 2nd Edition, 2012.	
Reference Books		
1.	Jatin Pandey, Darshana Pathak , Geographic Information System, The Energy and Resources Institute, TERI , 2013	
2.	Kang-Tsung Chang, **I**ntroduction to Geographic Information Systems, McGraw-Hill Higher Education, 6 edition, 2011	
3.	Basudeb Bhatta, Remote Sensing and GIS, Oxford; Second edition, 2011.	
Recommended by Board of Studies		5-3-2016
Approved by Academic Council	No. 40th	Date 18-3-2016

SWE 4005	Internet of Things	L	T	P	J	C
		2	**0**	0	4	3
Pre-requisite	**SWE3001**	**Syllabus version**				
		v. 1.0				

Course Objectives:

1. To understand fundamentals of Internet of things and its design aspects
2. To compreherd communication models with cloud environment.
3. To develop design thinking skills to new IoT based prototypes for real life applications.

Expected Course Outcome:

1. Design logical and physical structure of Internet of Things.
2. Develop the communication system and protocols for implementing Internet of Things
3. Use virtualization techniques for Internet of things.
4. Configure IOT devices..
5. Create or design functional model specification for Internet of Things based on domain specification
6. Design an Internet of Things application based on domain specification and real life applications using Internet of Things.
7. Identify level of domain specification
8. Understand Interactive products Development.

Module:1	Introduction to Internet of Things	**5 hours**

Introduction - Definition & Characteristics of IoT - Physical Design of IoT - Things in IoT - IoT Protocols, Logical Design of IoT - IoT Functional Blocks - IoT Communication Models - IoT Communication APIs, IoT Enabling Technologies

Module:2	IoT Levels & Deployment Templates	**5 hours**

Components in internet of things: Control Units – Sensors – Communication modules – Power Sources- IoT and M2M: Introduction - M2M - Difference between IoT and M2M - SDN and NFV for IoT - Software Defined Networking - Network Function Virtualization.

Module:3	IoT System Management with NETCONF-YANG	**6 hours**

Need for IoT Systems Management, Simple Network Management Protocol (SNMP), Limitations of SNMP, Network Operator Requirements, NETCONF, YANG, IoT Systems Management with NETCONF-YANG- Developing Internet Of Things -IoT Design Methodology

Module:4	**Domain Specific IoTs**	**6 hours**

Home Automation – Smart Cities – Environment – Health & Lifestyle Case Studies Illustrating IoT Design:Home Automation – Smart Lighting – Home Intrusion Detection, Cities – Smart Parking, Environment – Weather Monitoring System – Weather Reporting Bot – Air Pollution

Monitoring – Forest Fire Detection		
Module:5	**IoT Physical Devices and Endpoints**	**6 hours**
IoT Device – Basic building blocks of an IoT Device – Exemplary Device: Raspberry Pi – About the Board – Linux on Raspberry Pi – Raspberry Pi Interfaces – Serial – SPI – I2C – Programming Raspberry Pi – Other IoT Devices		
Module:6	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**30 hours**

Text Book(s)

1.	–Internet of things – Hands on approach‖ – ArshdeepBahga, Vijay Madisetti, Universities Press, 2015

Reference Books

1. Adrian McEwen & Hakim Cassimally, Designing the Internet of Things, Wiley, 2013
2. Samuel Greengard, The Internet of Things, MIT Press Essential Knowledge series, 2015
3. Donald Norris, The Internet of Things: Do-It-Yourself at Home Projects for Arduino, Raspberry Pi and BeagleBone Black, MCgraw Hill, 2015
4. Olivier Hersent, David Boswarthick, Omar Elloumi, The Internet of Things: Key Applications and Protocols, Wiley, 2012.

	Recommended by Board of Studies		5-3-2016	
	Approved by Academic Council	No. 40th	Date	18-3-2016

SWE 4006	Real Time Systems	L	T	P	J	C
		2	**0**	0	4	**3**
Pre-requisite	**SWE 3001**	**Syllabus version**				
		v. 1.20				

Course Objectives:

1. To provide a broad introduction to real time systems and their programming.
2. To explain and apply the fundamental concepts and terminology of real-time systems.
3. To bring students into the position to analyze and design real-time systems

Expected Course Outcome:

1. Understand the specific aspects of real-time systems
2. Understand main problems of the design of real-time systems and know some solutions
3. Will be able to use formal reasoning about real-time systems
4. Design real time models which includes temporal accuracy, permanence and idempotency
5. Design real time operating systems which enhances communication and task management
6. Configure commercial real time operating systems
7. Identify real time scheduling algorithm for design diversity, maintainability

Module:1	**INTRODUCTION**	**4 hours**

The Real Time Environment, Computer System Real time, Functional Requirements, Temporal requirements, Dependability requirements, Classification of real-time systems, Embedded real systems, plant automation system, multimedia systems, Examples.

Module:2	**REAL TIME MODELS**	**8 hours**

Real time model, model outline, component state, the message concept, component interfaces, gateway component, Linking interface specification, component integration, Temporal relations, QoS framework, QoS models.- **REAL TIME SYSTEMS PERFORMANCE-**Real time images, real time objects, temporal accuracy, permanence and idempotency, determinism, Dependability, basic concepts, information security, fault tolerance, robustness, Real time communications in LAN, RT Communication Over Packet Switched Networks

Module:3	**REAL TIME OPERATING SYSTEMS**	**4 hours**

Real time operating systems – inter component communication, task management, the dual role of time, inter task interactions, process input / output, error detection,

Module:4	**SCHEDULING REAL TIME TASKS**	**9 hours**

Real time scheduling – scheduling problem, worst case execution time, static scheduling, dynamic scheduling, alternative scheduling strategies-Real time System Design: System design – design

phases, design styles, safety analysis and standards, design diversity, maintainability. - **REAL TIME SYSTEM DESIGN -** System design – design phases, design styles, safety analysis and standards, design diversity, maintainability			
Module:5	**COMMERCIAL REAL TIME OPERATING SYSTEMS**	**3 hours**	
Time services, features of real time OS, Unix based real time OS, windows based real time OS, Linux based real time OS, benchmarking Real time systems, Applications in RT databases, concurrency control in RT databases and commercial RT databases			
Module:6	**Contemporary issues**	**2 hours**	
	Total Lecture hours:	**30 hours**	

Text Book(s)

1.	Kopetz, Hermann,Real-time systems: design principles for distributed embedded applications. Springer Science & Business Media, 2011.

Reference Books

1. Laplante, Phillip A., and Seppo J. Ovaska. Real-time systems design and analysis: tools for the practitioner. John Wiley and Sons, 2011.
2. Liu, Fan, Ajit Narayanan, and Quan Bai. "Real-time systems." (2000).
3. Krishna, C. Mani. Real-Time Systems. John Wiley & Sons, Inc., 1999.
4. Liu, Jane WS. "Real-time systems. 2000."
5. Rajib Mall, "Real Time Systems: Theory and Practice," Pearson, 2008.
6. C. Siva Ram Murthy and G. Manimaran, Resource Management in Real time Systems and Networks, MIT Press, March 2001

	Recommended by Board of Studies		12-8-2017	
	Approved by Academic Council	No. 47th	Date	5-10-2017

SWE4007	Storage Technologies	L	T	P	J	C
		3	0	0	0	3
Pre-requisite	SWE3001	Syllabus version				
		v.1.0				

Course Objectives:

1. To provide an understanding of guidelines, principles, and architecture used in storage technology.
2. To provide an insight into the technologies in storage management there by presenting the end user with through knowledge in designing secure storage system.

Expected Course Outcome:

1. Upon Completion of the course, the students will be able toIdentify each component and technologies implemented in storage infrastructures
2. Evaluate storage architectures; understand logical and physical components of a storage infrastructure including storage subsystems, RAID and Intelligent storage systems
3. Explain storage networking technologies such as FC SAN, NAS, IP-SAN, FCoE and data archival solution – CAS
4. Recognise different storage virtualization technologies and their benefits
5. Comprehend and articulate business continuity solutions, including, backup technologies, and local and remote replication solutions
6. Define information security, and storage security domains
7. Identify parameters of managing and monitoring storage infrastructure and describe common storage management activities and solutions
8. Apply the storage technology principles and design for various applications

Module:1	Storage Systems	6 hours

Storage Evolution and Data Center infrastructure. Host components, Connectivity, Storage, and Protocols. Components of a disk drive, physical disk and factors affecting disk drive performance. RAID level performance and availability considerations.

Module:2	Direct Attached Storage	6 hours

Direct Attached Storage (DAS)architecture, Storage Area Network (SAN) attributes, components, topologies, connectivity options and zoning. FC protocol stack, addressing, flow control, and classes of service.

Module:3	Networked Attached Storage	6 hours

Networked Attached Storage (NAS)components, protocols, IP Storage Area Network (IP SAN) iSCSI, FCIP and FCoE architecture. Content Addressed Storage (CAS) elements, storage, and retrieval processes

Module:4	Storage Virtualization	6 hours

Forms of Virtualization Memory Virtualization Network Virtualization Virtual SAN (VSAN) Server Virtualization Storage Virtualization Types of Storage Virtualization.

Module:5	Business Continuity	6 hours
Backup designs, architecture, topologies, and technologies in SAN and NAS environments. Local and Remote replication using host and array based replication technologies such as Synchronous and Asynchronous methods.		
Module:6	**Storage Security and Management**	**6 hours**
Securing the Storage Infrastructure - Storage Security Framework -Risk Triad -Assets -Threats -Vulnerability - Storage Security Domains Securing the Application Access Domain - Securing the Management Access Domain - Securing Backup, Recovery, and Archive (BURA)		
Module:7	**Storage Management Activities**	**7 hours**
Storage Management Activities -Availability management -Capacity management Performance management -Security Management -Reporting-Storage Management Examples Storage Infrastructure Management Challenges		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1. Somasundaram Gnanasundaram, Alok Shrivastava, Information Storage and Management, Wiley Publishing Inc, 2nd Edition ,2012

Reference Books

1. Data Storage Networking: Real World Skills for the CompTIA Storage+ Certification and Beyond Nigel Poulton John Wiley & Sons, 2014
2. Storage Networks Explained Ulf Troppens, Rainer Erkens, Wolfgang Muller-Friedt, Rainer Wolafka, Nils HausteinJohn Wiley & Sons, 24-Aug-2011
3. Securing Storage: A Practical Guide to SAN and NAS Security Himanshu Dwivedi ,Prentice Hall ,2012.

Recommended by Board of Studies		5-3-2016	
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE4008	High Performance Computing	L	T	P	J	C
		3	0	0	0	3
Pre-requisite	**SWE3001**	**Syllabus version**				
		v. 1.0				

Course Objectives:

1. To evaluate and compare the architectural features of the state of the art high performance commodity hardware platforms.
2. To study parallel algorithm design and programming issues for HPC systems.
3. To apply program optimization techniques to accelerate applications on the new high performance computing devices.

Expected Course Outcome:

1. Describe the overview and analyze the performance metrics of high performance parallel architectures
2. Illustrate the various High Performance Computing Paradigms.
3. Design High Performance Computing Applications
4. Develop various High Performance Computing applications using modern job scheduling tools
5. Analyze and measure the performance of high performance applications
6. Understand and Explore the various compiler optimization for HPC applications
7. Identify the emerging trends in high performance computing
8. Analyze and Implement current distributed Computing research literature

Module:1	High-Performance Parallel Architectures - On-Chip Instruction Level Parallelism:	8 hours

Pipelining- Basic concepts, instruction and arithmetic pipeline, data hazards, control hazards, and structural hazards, techniques for handling hazards. Pipeline optimization techniques. Compiler techniques for improving performance. Instruction-level parallelism: basic concepts, techniques for increasing ILP, superscalar, super-pipelined and VLIW processor architectures. Array and vector processors, data flow computers, reduction computer architectures, systolic architectures.

Module:2	Modern High-Performance Architectures:	5 hours

Multiprocessor architecture-Centralized shared-memory architecture - synchronization, memory consistency- Distributed shared-memory architecture, Cluster computers, Grids, Clouds, Many-Core Architecture.

Module:3	System Software Stack and Supercomputing Infrastructure:	5 hours

Storage, Distributed and Parallel File System, Parallel I/O, Interconnection network, System Software Stack, System Management and Monitoring Software, Supercomputing Infrastructure

Module:4	Design Issues in High Performance Computing:	5 hours
Synchronization, Scheduling, Job Allocation, Job Partitioning, Dependency Analysis, Mapping Parallel Algorithms onto Parallel Architectures, Bandwidth Limitations, Latency Limitations, Latency Hiding/Tolerating Techniques and their limitations.		
Module:5	**Performance Evaluation:**	**6 hours**
Performance Analysis of Parallel Algorithms - Basics of Performance Evaluation, Sources of Parallel Overhead, Speedup Performance Laws, Scalability metric, Performance Measurement Tools, Identifying performance bottlenecks, Restructuring applications for deep memory hierarchies, Partitioning applications for heterogeneous resources, Using existing libraries, tools and frameworks.		
Module:6	**Compiler Optimization Techniques:**	**6 hours**
Granularity and Partitioning, Locality: temporal/spatial/stream/kernel. Compiler Transformations for Parallel Computers- Issues in Compiler Transformations, Dependence Analysis, Data Dependency Reduction. Data flow. Loop reordering.		
Module:7	**Power-Aware Computing and Current Trends in HPC:**	**8 hours**
Power-aware Processing Techniques, Power-aware Memory Design, Power-aware Interconnect Design, Software Power Management, Petascale Computing, Optics in Parallel Computing, Quantum Computers, Recent developments in Nanotechnology and its impact on HPC.		
Module:8	**Contemporary issues .**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1.	Kai Hwang, Advanced Computer Architecture: Parallelism, Scalability, Programmability, Third Edition, McGraw Hill, 2015.

Reference Books

1.	**John Levesque, Gene Wagenbreth,** High Performance Computing: Programming and Applications, Chapman & Hall/CRC, First Edition, 2010.
2.	Jeffrey S. Vetter, Chapman and Hall, Contemporary High Performance Computing: From Petascale to Exascale, CRC, 2013.
3.	David A. Bader, Chapman & Hall, Petascale Computing: Algorithms and Applications, CRC Computational Science Series, 2008

Recommended by Board of Studies	5-3-2016		
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE4009	Linux Programming	L	T	P	J	C
		3	0	2	0	4
Pre-requisite	SWE3001	**Syllabus version**				
		v. 1.0				

Course Objectives:

1. Demonstrate the development philosophy of Linux
2. Create shell scripts for any service.
3. Maintain the source code and docs with standard repositories

Expected Course Outcome:

1. Understand Linux Programming Methods
2. Write shell scripting for any task automation
3. Debug the program using tools for maintaining coding standards
4. Analyse FHS and Semaphores
5. Use digital code repositories for source code and documents maintenance
6. Apply the data management and development tools
7. Explore the process management structure

Module:1	Introduction to Linux Programming	5 hours

An introduction to UNIX, Linux and GNU project, FSF, Linux distributions, Programming Linux, Compilers, Editors, Linux development model, cathedral and Bazzar, Linux community, Standards for Linux and uniqueness of Linux.

Module:2	Shell Programming	6 hours

Types of shells, Pipes and redirection, Shell Syntax, Writing shell scripts for frequent backups, log monitoring, history monitoring and system parameters logging, user management and system management

Module:3	Debugging	5 hours

General debugging techniques, debugging with gdb, starting gdb, running a program, understanding stacktrace and breakpoints, more debugging tools, assertions and memory debugging, using gdb for the shell scripts and programming languages, graphic debugging tools

Module:4	Environment variables and Working with files	7 hours

Environment variables for time date, files, user and host logging , File system Hierarchy Standard,

System calls and device drivers, Library functions, Low level file access, standard I/O library, Formatted I/O, File and directory maintenance, Scanning directories, errors, /proc file system, advanced topics, fcntl, mmap		
Module:5	**Terminals and Managing text based screens with curses**	**7 hours**
Talking to the terminal, termios structure, terminal output and key strokes, curses terminology and concepts, the screen, the keyboard, the windows and subwindows, colors, pads and the CD collection application.		
Module:6	**Data Management and development tools**	**6 hours**
Managing memory, File locking, databases, The CD application, The make command and make files, Source code control, Writing a manual page, distributing software, package formats and environments.		
Module:7	**Processes, IPC and Semaphores**	**7 hours**
Process structure and signals, Process pipes, parent and child processes, named pipes, Semaphores ,Shared memory, Message queues, IPC status commands		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1. Robert Love, Linux System Programming: Talking Directly to the Kernel and C Library 2e, O Reilly media, 2013.

Reference Books

1. Neil Mathew, Richard Stones, Beginning Linux Programming, 4e, Wiley Publications, 2008
2. John Masters, Richard Blum, Professional Linux Programming, Wiley Publications ,2007

List of Challenging Experiments (Indicative)

1	Installation of Linux and Windows in a single machine with various partitioning options	
2	Create shell scripts for user management, system management, backup and restore processes	
3	Create shell scripts for process management and memory management Create shell script to use curses	
4	Configure GRUB/LILO using scripts	
5	Write a script to monitor the system logs and ensure security	
6	Create shell script for managing ACL policies with files	

7	Debug the user created and standard shell scripts	
8	Create SVN to maintain the project documents	
9	Create a gitub/gitlab account to maintain your PBL source code for collaborative development	
10	With the help of uck produce your own kernel and perform installation on a laptop / desktop	
11	Modify the grub configuration to have your own custom modules	
12	Write into the display configuration file of Linux for booting with custom modules	
	Total Laboratory Hours	30 hours

Recommended by Board of Studies		5-3-2016	
Approved by Academic Council	No. 40th	Date	18-3-2016

SWE4010	Artificial Intelligence	L	T	P	J	C
		3	0	0	4	4
Pre-requisite		**Syllabus version**				
		Version. 1.0				

Course Objectives:

- To understand the basics of Artificial Intelligence.
- To improve problem solving techniques, knowledge representation and reasoning systems capability.
- To gain knowledge for developing an Intelligent agent

Expected Course Outcome:

On completion of this course, student should be able to

1. Learn various Artificial Intelligence techniques and their areas of applications.
2. Solve various practical problems using Artificial Intelligence techniques.
3. Understand the problem space and searching methods especially heuristic search
4. Equip with different data representations and languages for artificial intelligent systems.
5. Enhance the reasoning ability using Predicate Logic
6. Learn to take decision under uncertainties
7. Develop skills for planning and learning.
8. Develop applications using NLP technique

Module:1	**Introduction**	**6 hours**

Foundation of AI- History-Intelligent Agents –Agent and environment

Module:2	**Problem Solving**	**8 hours**

Solving problems by searching- Uninformed search- BFS,DFS, Uniform cost search Informed search- Best First search, A* search, Local search- Hill climbing, Two player games

Module:3	**Knowledge Representation**	**4 hours**

Rule based system, Semantic net, Reasoning in Semantic Net, Frames and slots

Module:4	**Reasoning**	**8 hours**

Propositional Logic, Reasoning usi ng First order logic, Forward and backward reasoning, Unification, Resolution.

Module:5	**Uncertainty-Probabilistic Reasoning**	**6 hours**

Prior and Posterior Probabilities - Bayes' Theorem – Bayesian Network- Probabilistic reasoning over time- Inference in temporal model- Hidden Markov Model

Module:6	**Planning and Learning**	**6 hours**

Representation for planning-Partial order Planning – Total order Planning –Learning – Learning - Forms of learning – Choosing the best hypothesis , Classification and regression

Module:7	**Natural Language Processing**	**5 hours**

Language models- Model evaluation- Text classification-Information retrieval, Page- Rank algorithm, Information extraction

Module:8	**Contemporary issues**	**2 hours**

	Total Lecture hours:	**45 hours**
Text Book(s)		
1. Stuart J . Russell and Peter Norvig, Artificial Intelligence: A Modern Approach, Third Edition, PHI, 2015		
Reference Books		
1. Elaine Rich and Kevin Knight, Artificial Intelligence, Third Edition, Tata McGraw Hill, 2008 2. Patrick Henry Winston, Artificial Intelligence, Third Edition, Addison Wesley, 2011		
Recommended by Board of Studies	02 03-2019	
Approved by Academic Council	No:54th	Date :14-03-2019

SWE4011	Game Programming	L	T	P	J	C
		3	0	2	0	4

Pre-requisite	Nil	Syllabus version
		v.1.0

Course Objectives:

- To understand game logic, design, development, processes and mechanics
- To build and then integrate technologies such as multimedia, artificial intelligence, and physics modelling into a cohesive, interactive game application.
- To learn and use software engineering, team project management and techniques currently used in the game industry

Expected Course Outcome:

On completion of the course the students will be able to

1. Develop, debug, and modify code to meet design specifications for games.
2. Develop, test, and evaluate procedures of the creation, design and development of games.
3. Design unique gaming environments, levels and characters.
4. Create games by applying programming concepts.
5. Create and produce digital components, games and documentation using a variety of computer platforms.
6. Choose game strategies and patterns based on an analysis of past and present trends.
7. Contribute as an individual and a member of a team and provide leadership as required.

Module:1	Introduction to Game Programming	2 hour

Overview of game programming, Structure of a typical game team, game industry, game engine history.

Module:2	Game Engine Architecture	8 hours

Real Time Game Architecture, Engine Support: Subsystem Start-Up and Shut-Down, Memory Management, Containers and Strings; Resource Management: File System, Resource Manager.

Module:3	Graphics for game programming	8 hours

Graphics Device Management, The Rendering Engine: The Rendering Pipeline, Lighting and Global Illumination, Sprites, Tile-Based Graphics and Scrolling, GUI programming for games

Module:4	Artificial Intelligence for Interactive Environments	8 hours

Why Artificial Intelligence for Games, AI methods in gaming: Tree search, Reinforcement learning, Path finding algorithms: Dijkstra's algorithm, A* algorithm, D* Algorithm and navigation meshes.

Module:5	**Game Physics**	**8 hours**
Physics based modeling, Rigid Body Dynamics, Integrating a Physics Engine into the Game; Collision detection: Object boundaries, Sphere algorithms, Cuboid algorithms, Point algorithms, Line algorithms.		
Module:6	**Game design**	**5 hours**
Game design, Game genres, modes, and perspectives, scripting, audio engineering, Sound and Music, level design, render threading		
Module:7	**Project management in game development**	**4 hours**
Game project management, Game design documentation, Rapid prototyping and game testing		
Module:8	**Contemporary issues**	**2 hour**
Recent trends in game industry		
	Total Lecture hours:	**45 hours**

Text Book(s)

1. Game Engine Architecture, 3rd Edition, Jason Gregory, A K Peters, 2019

Reference Books

1. Yannakakis GN, Togelius J. Artificial intelligence and games. New York: Springer; 2018 Feb 17.
2. Akenine-Moller T, Haines E, Hoffman N. Real-time rendering. AK Peters/CRC Press; 2018 Jul 20.
3. Best of Game Programming Gems, Mark DeLoura, Course Technology, Cengage Learning, 2014
4. Real-Time Collision Detection, Christer Ericson, Morgan Kaufmann, 2005
5. 4. XNA Game Studio 4.0 Programming. Tom Miller and Dean Johnson, Addison-Wesley Professional, 2010
6. Game Coding Complete, Mike McShaffry and David Graham, Fourth Edition, 2012 Cengage Learning PTR
7. Beginning Game Programming, Jonathan S. Harbour, Cengage Learning PTR; 4th edition, 2014
8. Fundamentals of Game Design, 3rd Edition, Ernest Adams, New Riders; 2013
9. Game Design Foundations, Second Edition, Roger E. Pedersen, Jones & Bartlett Learning; 2009
10. Level Up! The Guide to Great Video Game Design, 2nd Edition, Scott Rogers, Wiley 2014

List of Challenging Experiments (Indicative)

1. Create a 2D game named –Flappy Bird‖ which can fly the bird as far as the player can without hitting a pipe kind of border on its left and right side. Once the player reaches a particular level, winning note should be displayed.

2. Create a 3D game name –Ogre‖, where a player has to travel and reach the goal inside a maze without hitting the patrols.

3.	Create a 3D Bowling game which has 6 pins and a ball by applying the collision detection techniques and calculate the score accordingly.
4.	Create a game component using MAYA/ Blender software.
5.	Create a 2D game by extending exercise 1 with multiple levels.
6.	Create a tile-based game which allows the user to move the player over the tiles.
7.	Develop a VR game which can trigger the player movements using the click events of VR google card.
8.	Develop a Tic-Tac-Toe game utilizing only the UI components.
9.	Develop a Tetris game. It is a single player game where the player has to manipulate blocks that fall down from the top of the screen in such a way that rows on the bottom are filled. When a row is filled, it disappears and the player receives points.
10.	Develop a 2048 game. It is a single-player sliding block puzzle game. The game's objective is to slide numbered tiles on a grid to combine them to create a tile with the number 2048.

Total Laboratory Hours	30 hours	
Recommended by Board of Studies	02 03-2019	
Approved by Academic Council	No:54th	Date :14-03-2019

SWE4012	Machine Learning	L	T	P	J	C
		3	0	2	0	4
Pre-requisite		**Syllabus version**				
		Version. 1.0				

Course Objectives:

- To make the scholars familiar with different forms of learning algorithms, Regression models and Classification Methods
- To enable the learners with an in-depth understanding of Graphical Models, and Ensemble Methods with emphasis on complex problem-solving techniques
- To empower the scholars the knowledge about Computational Learning Theory, Unsupervised Learning with a specific focus on practical, real-world issues.

Expected Course Outcome:

On completion of this course, the student should be able to

1. Exhibit knowledge of the fundamental elements and concepts related to machine learning algorithms
2. Ability to identify sundry means of choosing apposite Computational Learning Model and implementing the model successfully
3. Use and apply the suitable Regression Analysis for various type of learning problems
4. Develop the Classification Methods and suitable solutions for problems that deal with small and large dataset
5. Apply important methods in Graphical Models for various real-world problems
6. Apply the knowledge and skills for solving realistic and logical issues using Ensemble Learning Methods
7. Develop improved machine learning methods, related unsupervised learning and computing models and programming framework for practical applications
8. Implement various solutions with the help of machine learning approaches for achieving appropriate decisions for pragmatic everyday problems

Module:1	**Basics**	6 **hours**

Introduction to machine learning - Types of machine learning, Supervised learning, Unsupervised learning, Machine learning process, Basics of probability theory and Linear algebra and other Preliminaries

Module:2	**Computational Learning Theory**	6 **hours**

Concept learning, Version Spaces, Finite and Infinite Hypothesis Spaces, PAC Learning, VC Dimension, Bias/Variance Trade-offs.

Module:3	**Regression Analysis**	**6 hours**

Linear regression, Polynomial Regression, Stepwise Regression, Ridge regression, Lasso, ElasticNet Regression

Module:4	**Classification Methods**	**7 hours**
Linear Discriminant Analysis, Logistic regression, k-Nearest Neighbors Method, Naïve Bayes Method, Large margin classification, Support Vector Machines, Classification and Regression Trees		
Module:5	**Graphical Models**	**6 hours**
Bayesian Belief Networks, Markov Random Fields, Hidden Markov Models, Exact inference methods, Approximate inference methods.		
Module:6	**Ensemble Learning**	**6 hours**
Boosting - Adaboost, Gradient Boosting; Bagging - Simple methods, Random Forest**,** Stacking		
Module:7	**Unsupervised Learning**	**6 hours**
Introduction to clustering, Hierarchical: AGNES, DIANA, Partitioned: K-means clustering, K-Mode Clustering, Expectation Maximization, Gaussian Mixture Models		
Module:8	**Deep Learning Models**	**2 hours**
Guest Lecture		
	Total Lecture hours:	**45 hours**

Text Book(s)

1. Christopher M. Bishop, Pattern Recognition and Machine Learning, Springer, 2013.

Reference Books

1. Kevin P. Murphy, Machine Learning – A Probabilistic Perspective, MIT Press (MA), 2014.
2. T. Hastie, R. Tibshirani, and J. H. Friedman. The Elements of Statistical Learning: Data Mining, Inference and Prediction.2nd Edition, Springer, 2008.
3. Mitchell, Tom. Machine Learning. McGraw-Hill, 2013.
4. Mehryar Mohri, Afshin Rostamizadeh, Ameet Talwalkar "Foundations of Machine Learning‖, MIT Press, 2012.
5. Stephen Marsland, Machine Learning: An Algorithmic Perspective, Second Edition, CRC Press, 2014

CHY1701	Engineering Chemistry	L	T	P	J	C
		3	0	2	0	4
Pre-requisite	Chemistry of 12th standard or equivalent	Syllabus version				
		1.1				

Course Objectives:

- To impart technological aspects of applied chemistry
- To lay foundation for practical application of chemistry in engineering aspects

Expected Course Outcome:

1) Recall and analyze the issues related to impurities in water and their removal methods and apply recent methodologies in water treatment for domestic and industrial usage
2) Evaluate the causes of metallic corrosion and apply the methods for corrosion protection of metals
3) Evaluate the electrochemical energy storage systems such as lithium batteries, fuel cells and solar cells, and design for usage in electrical and electronic applications
4) Assess the quality of different fossil fuels and create an awareness to develop the alternative fuels
5) Analyze the properties of different polymers and distinguish the polymers which can be degraded and demonstrate their usefulness
6) Apply the theoretical aspects: (a) in assessing the water quality; (b) understanding the construction and working of electrochemical cells; (c) analyzing metals, alloys and soil using instrumental methods; (d) evaluating the viscosity and water absorbing properties of polymeric materials

Module:1	Water Technology	

Hardness of water - hardness causing impurities, pH, DO, TDS, COD and BOD in water; Estimation of hardness by EDTA method-numerical problems. Boiler troubles - scale, sludge, priming, foaming, caustic embrittlement and boiler corrosion; Internal conditioning – Phosphate and calgon conditioning methods

Module:2	Water Treatment	8 hours

Water treatment for Industrial purpose: External softening methods: Lime Soda process-numerical problems, Zeolite process and ion exchange including mixed bed ion exchange processes. Steps involved in treatment of water for municipal supply – Water purification fordomestic purpose - Activated carbon filtration, UV treatment, Ozonolysis, Reverse osmosis.

Module:3	Corrosion	6 hours

Types and mechanism – dry and wet corrosion; Forms of corrosion [Differential aeration, pitting, Galvanic and stress corrosion cracking]; Factors affecting corrosion

Module:4	Corrosion Control	4 hours

Corrosion control methods: Inhibitors – anodic and cathodic and their action; Cathodic protection

– sacrificial anodic and impressed current protection methods. Corrosion protection coatings: galvanizing and tinning; electroplating-processes and typical applications; Advanced coating processes – Basic concepts of PVD and CVD

Module:5	**Electrochemical Energy Systems**	**6 hours**

Basic concepts of cells and batteries-nominal voltage, operating voltage, capacity, self-discharge, depth of discharge, energy density, service life, shelf life. Working and applications of primary cells - Alkaline cells -and Li-primary cells.
Secondary cells and batteries - Ni-MH cells; Rechargeable lithium cells – chemistry and applications. Fuel cells – Electrochemistry of a H_2–O_2 fuel cell, Basics of solid oxide fuel cells-applications

Module:6	**Fuels and Combustion**	**8 hours**

Calorific value - Definition of LCV, HCV. Measurement of calorific value using bomb calorimeter and Boy‘s calorimeter including numerical problems. Combustion of fuels - minimum quantity of air by volume and by weight-Numerical problems. Knocking and chemical structure, octane number and cetane number and their importance;
Biodiesel-synthesis, advantages and commercial applications

Module:7	**Polymers**	**6 hours**

Thermoplastic & Thermo setting resins – comparative properties. Properties and engineering applications of ABS, PVC,Teflon and Bakelite. Compression, injection, extrusion, Transfer moulding methods of plastics.
Conducting polymers: Intrinsic, extrinsic and doped polymers - Polyacetylene-mechanism of conduction- Applications of conducting polymers in LEDs, Mobile phones

Module:8	**Contemporary issues:**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1. 1. Sashi Chawla, A Text book of Engineering Chemistry, Dhanpat Rai Publishing Co., Pvt. Ltd., Educational and Technical Publishers, New Delhi, 3rd Edition, 2015.
2. O.G. Palanna, McGraw Hill Education (India) Private Limited, 9th Reprint, 2015.
3. B. Sivasankar, Engineering Chemistry 1st Edition, Mc Graw Hill Education (India), 2008

Reference Books

1. 1. O.V. Roussak and H.D. Gesser, Applied Chemistry-A Text Book for Engineers and Technologists, Springer Science Business Media, New York, 2nd Edition, 2013.
2. S. S. Dara, A Text book of Engineering Chemistry, S. Chand & Co Ltd., New Delhi, 20th Edition, 2013.

Mode of Evaluation: Internal Assessment (CAT, Quizzes, Digital Assignments) & FAT

List of Challenging Experiments (Indicative)

	Experiment title	Hours
1.	Estimation of Dissolved Oxygen by Winkler‘s Method	1 h 50 min
2.	Softening of Water through Zeolite Resin – Assessment of Total Hardness using EDTA Method	1 h 50 min

3.	Water Preservation through Smart Materials	1 h 50 min
4.	Construction and Working of an Electrochemical Cell	1 h 50 min
5.	Irrigation Water - Sulphate ion Analysis by Conductometry	1 h 50 min
6.	Estimation of Calcium Hardness in Water by Flame Photometry	1 h 50 min
7.	Estimation of Nickel in a Ni-plated Material for Corrosion Protection by Colorimetry	1 h 50 min
8.	Analysis of Iron in Steel by Potentiometric Method	1 h 50 min
9.	Determination of Aromatic Content in Diesel by Aniline Point Measurement	1 h 50 min
10.	Engineering Polymers - Viscosity and Molecular Weight Analysis	1 h 50 min
11.	Lab Scale Production of Biodiesel from Plant Seeds (demo experiment)	3 hours
	Total Laboratory Hours	18 hours

Mode of Evaluation: Viva-voce and Lab performance & FAT			
Recommended by Board of Studies	12.08.2017		
Approved by Academic Council	46th ACM	Date	24-8-17

CSE1001	PROBLEM SOLVING AND PROGRAMMING	L	T	P	J	C
		0	0	6	0	3
Pre-requisite	NIL	Syllabus version				
		1.0				

Course Objectives:

1. To develop broad understanding of computers, programming languages and their generations
2. Introduce the essential skills for a logical thinking for problem solving
3. To gain expertise in essential skills in programming for problem solving using computer

Expected Course Outcome:

1. Understand the working principle of a computer and identify the purpose of a computer programming language.
2. Learn various problem solving approaches and ability to identify an appropriate approach to solve the problem
3. Differentiate the programming Language constructs appropriately to solve any problem
4. Solve various engineering problems using different data structures
5. Able to modulate the given problem using structural approach of programming
6. Efficiently handle data using flat files to process and store data for the given problem

List of Challenging Experiments (Indicative)

1	Steps in Problem Solving Drawing flowchart using yEd tool/Raptor Tool	4 Hours
2	Introduction to Python, Demo on IDE, Keywords, Identifiers, I/O Statements	4 Hours
3	Simple Program to display Hello world in Python	4 Hours
4	Operators and Expressions in Python	4 Hours
5	Algorithmic Approach 1: Sequential	4 Hours
6	Algorithmic Approach 2: Selection (if, elif, if.. else, nested if else)	4 Hours
7	Algorithmic Approach 3: Iteration (while and for)	6 Hours
8	Strings and its Operations	6 Hours
9	Regular Expressions	6 Hours
10	List and its operations	6 Hours
11	Dictionaries: operations	6 Hours
12	Tuples and its operations	6 Hours
13	Set and its operations	6 Hours
14	Functions, Recursions	6 Hours
15	Sorting Techniques (Bubble/Selection/Insertion)	6 Hours
16	Searching Techniques : Sequential Search and Binary Search	6 Hours

17	Files and its Operations	6 Hours
	Total hours:	**90 hours**

Text Book(s)	
1.	John V. Guttag., 2016. Introduction to computation and programming using python: with applications to understanding data. PHI Publisher.
Reference Books	
1.	Charles Severance.2016.Python for everybody: exploring data in Python 3, Charles Severance.
2.	Charles Dierbach.2013.Introduction to computer science using python: a computational problem-solving focus. Wiley Publishers.

Mode of Evaluation: **PAT / CAT / FAT**			
Recommended by Board of Studies	04-04-2014		
Approved by Academic Council	No. 38	Date	23-10-2015

CSE1002	PROBLEM SOLVING AND OBJECT ORIENTED PROGRAMMING	L	T	P	J	C
		0	0	6	0	3
Pre-requisite	**Nil**	**Syllabus version**				
		1.0				

Course Objectives:

1. To emphasize the benefits of object oriented concepts.

2.To enable students to solve the real time applications using object oriented programming features

3.To improve the skills of a logical thinking and to solve the problems using any processing elements

Expected Course Outcome:

1. Demonstrate the basics of procedural programming and to represent the real world entities as programming constructs.
2. Enumerate object oriented concepts and translate real-world applications into graphical representations.
3. Demonstrate the usage of classes and objects of the real world entities in applications.
4. Discriminate the reusability and multiple interfaces with same functionality based features to solve complex computing problems.
5. Illustrate possible error-handling constructs for unanticipated states/inputs and to use generic programming constructs to accommodate different datatypes.
6. Validate the program against file inputs towards solving the problem.

List of Challenging Experiments (Indicative)

1.	**Postman Problem** A postman needs to walk down every street in his area in order to deliver the mail. Assume that the distances between the streets along the roads are given. The postman starts at the post office and returns back to the post office after delivering all the mails. Implement an algorithm to help the post man to walk minimum distance for the purpose.	10 hours

2.	**Budget Allocation for Marketing Campaign** A mobile manufacturing company has got several marketing options such as Radio advertisement campaign, TV non peak hours campaign, City top paper network, Viral marketing campaign, Web advertising. From their previous experience, they have got a statistics about paybacks for each marketing option. Given the marketing budget (rupees in crores) for the current year and details of paybacks for each option, implement an algorithm to determine the amount that shall spent on each marketing option so that the company attains the maximum profit.	15 hours
3.	**Missionaries and Cannibals** Three missionaries and three cannibals are on one side of a river, along with a boat that can hold one or two people. Implement an algorithm to find a way to get everyone to the other side of the river, without ever leaving a group of missionaries in one place outnumbered by the cannibals in that place.	10 hours
4.	**Register Allocation Problem** A register is a component of a computer processor that can hold any type of data and can be accessed faster. As registers are faster to access, it is desirable to use them to the maximum so that the code execution is faster. For each code submitted to the processor, a register interference graph (RIG) is constructed. In a RIG, a node represents a temporary variable and an edge is added between two nodes (variables) t1 and t2 if they are live simultaneously at some point in the program. During register allocation, two temporaries can be allocated to the same register if there is no edge connecting them. Given a RIG representing the dependencies between variables in a code, implement an algorithm to determine the number of registers required to store the variables and speed up the code execution	15 hours
5.	**Selective Job Scheduling Problem** A server is a machine that waits for requests from other machines and responds to them. The purpose of a server is to share hardware and software resources among clients. All the clients submit the jobs to the server for execution and the server may get multiple requests at a time. In such a situation, the server schedule the jobs submitted to it based on some criteria and logic. Each job contains two values namely time and memory required for execution. Assume that there are two servers that schedules jobs based on time and memory. The servers are named as Time Schedule Server and memory Schedule Server respectively. Design a OOP model and implement the time Schedule Server and memory Schedule Server. The Time Schedule Server arranges jobs based on time required for execution in ascending order whereas memory Schedule Server arranges jobs based on memory required for execution in ascending order	15 hours
6.	**Fragment Assembly in DNA Sequencing** DNA, or deoxyribonucleic acid, is the hereditary material in humans and almost all other organisms. The information in DNA is stored as a code made up of four chemical bases: adenine (A), guanine (G), cytosine (C), and	15 hours

	thymine (T). In DNA sequencing, each DNA is sheared into millions of small fragments (reads) which assemble to form a single genomic sequence (superstring). Each read is a small string. In such a fragment assembly, given a set of reads, the objective is to determine the shortest superstring that contains all the reads. For example, given a set of strings, 000, 001, 010, 011, 100, 101, 110, 111 the shortest superstring is 0001110100. Given a set of reads, implement an algorithm to find the shortest superstring that contains all the given reads.	
7.	**House Wiring** An electrician is wiring a house which has many rooms. Each room has many power points in different locations. Given a set of power points and the distances between them, implement an algorithm to find the minimum cable required.	10 hours
	Total Laboratory Hours	**90 hours**

Text Book(s)	
1.	Stanley B Lippman, Josee Lajoie, Barbara E, Moo, C++ primer, Fifth edition, Addison-Wesley, 2012.
2	Ali Bahrami, Object oriented Systems development, Tata McGraw - Hill Education, 1999.
3	Brian W. Kernighan, Dennis M. Ritchie , The C programming Language, 2nd edition, Prentice Hall Inc., 1988.

Reference Books	
1.	Bjarne stroustrup, The C++ programming Language, Addison Wesley, 4th edition, 2013
2.	Harvey M. Deitel and Paul J. Deitel, C++ How to Program, 7th edition, Prentice Hall, 2010
3.	Maureen Sprankle and Jim Hubbard, Problem solving and Programming concepts, 9th edition, Pearson Eduction, 2014.

Mode of assessment: **PAT / CAT / FAT**			
Recommended by Board of Studies	29-10-2015		
Approved by Academic Council	No. 39	Date	17-12-2015

ENG1901	Technical English - I	L	T	P	J	C
		0	0	4	0	2
Pre-requisite	Foundation English-II	**Syllabus Version**				
		1				

Course Objectives:

1. To enhance students‘ knowledge of grammar and vocabulary to read and write error-free language in real life situations.
2. To make the students‘ practice the most common areas of written and spoken communications skills.
3. To improve students‘ communicative competency through listening and speaking activities in the classroom.

Expected Course Outcome:

1. Develop a better understanding of advanced grammar rules and write grammatically correct sentences.
2. Acquire wide vocabulary and learn strategies for error-free communication.
3. Comprehend language and improve speaking skills in academic and social contexts.
4. Improve listening skills so as to understand complex business communication in a variety of global English accents through proper pronunciation.
5. Interpret texts, diagrams and improve both reading and writing skills which would help them in their academic as well as professional career.

Module:1	**Advanced Grammar**	**4 hours**

Articles, Tenses, Voice and Prepositions

Activity: Worksheets on Impersonal Passive Voice, Exercises from the prescribed text

Module:2	**Vocabulary Building I**	**4 hours**

Idioms and Phrases, Homonyms, Homophones and Homographs

Activity: Jigsaw Puzzles; Vocabulary Activities through Web tools

Module:3	**Listening for Specific Purposes**	**4 hours**

Gist, monologues, short conversations, announcements, briefings and discussions

Activity: Gap filling; Interpretations

Module:4	**Speaking for Expression**	**6 hours**

Introducing oneself and others, Making Requests & responses, Inviting and Accepting/Declining Invitations

Activity: Brief introductions; Role-Play; Skit.

Module:5	**Reading for Information**	**4 hours**

Reading Short Passages, News Articles, Technical Papers and Short Stories

Activity: Reading specific news paper articles; blogs

Module:6	**Writing Strategies**	**4 hours**
Joining the sentences, word order, sequencing the ideas, introduction and conclusion Activity: Short Paragraphs; Describing familiar events; story writing		
Module:7	**Vocabulary Building II**	**4 hours**
Enrich the domain specific vocabulary by describing Objects, Charts, Food, Sports and Employment. Activity: Describing Objects, Charts, Food, Sports and Employment		
Module:8	**Listening for Daily Life**	**4 hours**
Listening for statistical information, Short extracts, Radio broadcasts and TV interviews Activity: Taking notes and Summarizing		
Module:9	**Expressing Ideas and Opinions**	**6 hours**
Telephonic conversations, Interpretation of Visuals and describing products and processes. Activity: Role-Play (Telephonic); Describing Products and Processes		
Module: 10	**Comprehensive Reading**	**4 hours**
Reading Comprehension, Making inferences, Reading Graphics, Note-making, and Critical Reading. Activity: Sentence Completion; Cloze Tests		
Module: 11	**Narration**	**4 hours**
Writing narrative short story, Personal milestones, official letters and E-mails. Activity: Writing an E-mail; Improving vocabulary and writing skills.		
Module:12	**Pronunciation**	**4 hours**
Speech Sounds, Word Stress, Intonation, Various accents Activity: Practicing Pronunciation through web tools; Listening to various accents of English		
Module:13	**Editing**	**4 hours**
Simple, Complex & Compound Sentences, Direct & Indirect Speech, Correction of Errors, Punctuations. Activity: Practicing Grammar		
Module:14	**Short Story Analysis**	**4 hours**
‒The Boundary‖ by Jhumpa Lahiri Activity: Reading and analyzing the theme of the short story.		
	Total Lecture hours	**60 hours**

Text Book / Workbook	
1.	Wren, P.C.; Martin, H.; Prasada Rao, N.D.V. (1973–2010). *High School English Grammar & Composition*. New Delhi: Sultan Chand Publishers.
2	Kumar, Sanjay,; Pushp Latha. (2018) English Language and Communication Skills for Engineers, India: Oxford University Press.
Reference Books	
1.	Guptha S C, (2012) *Practical English Grammar & Composition*, 1st Edition, India: Arihant Publishers

2.	Steven Brown, (2011) Dorolyn Smith**,** *Active Listening **3**,* 3rd Edition, UK: Cambridge University Press.
3.	Liz Hamp-Lyons, Ben Heasley, (2010) *Study Writing*, 2nd Edition, UK: Cambridge University Pres.
4.	Kenneth Anderson, Joan Maclean, (2013) Tony Lynch, *Study Speaking*, 2nd Edition, UK: Cambridge, University Press.
5.	Eric H. Glendinning, Beverly Holmstrom, (2012) *Study Reading*, 2nd Edition, UK: Cambridge University Press.
6.	Michael Swan, (2017) *Practical English Usage* (Practical English Usage), 4th edition, UK: Oxford University Press.
7.	Michael McCarthy, Felicity O'Dell, (2015) *English Vocabulary in Use Advanced* (South Asian Edition), UK: Cambridge University Press.
8.	Michael Swan, Catherine Walter, (2012) *Oxford English Grammar Course Advanced*, Feb, 4th Edition, UK: Oxford University Press.
9.	Watkins, Peter. (2018) *Teaching and Developing Reading Skills: Cambridge Handbooks for Language teachers*, UK: Cambridge University Press.
10.	(*The Boundary by Jhumpa Lahiri)* URL: https://www.newyorker.com/magazine/2018/01/29/the-boundary?intcid=inline_amp

Mode of evaluation: Quizzes, Presentation, Discussion, Role play, Assignments and FAT

List of Challenging Experiments (Indicative)		
1.	Self-Introduction	**12 hours**
2.	Sequencing Ideas and Writing a Paragraph	**12 hours**
3.	Reading and Analyzing Technical Articles	**8 hours**
4.	Listening for Specificity in Interviews (Content Specific)	**12 hours**
5.	Identifying Errors in a Sentence or Paragraph	**8 hours**
6.	Writing an E-mail by narrating life events	**8 hours**
	Total Laboratory Hours	**60 hours**

Mode of evaluation: Quizzes, Presentation, Discussion, Role play, Assignments and FAT

Recommended by Board of Studies	08.06.2019	
Approved by Academic Council	55	Date: 13-06-2019

ENG1902	Technical English - II	L	T	P	J	C
		0	0	4	0	2
Pre-requisite	71% to 90% EPT score	**Syllabus Version**				
		1				

Course Objectives:

1. To acquire proficiency levels in LSRW skills on par with the requirements for placement interviews of high-end companies / competitive exams.
2. To evaluate complex arguments and to articulate their own positions on a range of technical and general topics.
3. To speak in grammatical and acceptable English with minimal MTI, as well as develop a vast and active vocabulary.

Expected Course Outcome:

1. Communicate proficiently in high-end interviews and exam situations and all social situations
2. Comprehend academic articles and draw inferences
3. Evaluate different perspectives on a topic
4. Write clearly and convincingly in academic as well as general contexts
5. Synthesize complex concepts and present them in speech and writing

Module:1	**Listening for Clear Pronunciation**	**4 hours**

Ice-breaking, Introduction to vowels, consonants, diphthongs.
Listening to formal conversations in British and American accents (BBC and CNN) as well as other _native' accents
Activity: Factual and interpretive exercises; note-making in a variety of global English accents

Module:2	**Introducing Oneself**	**4 hours**

Speaking: Individual Presentations
Activity: Self-Introductions, Extempore speech

Module:3	**Effective Writing**	**6 hours**

Writing: Business letters and Emails, Minutes and Memos
Structure/ template of common business letters and emails: inquiry/ complaint/ placing an order; Formats of Minutes and Memos
Activity: Students write a business letter and Minutes/ Memo

Module:4	**Comprehensive Reading**	**4 hours**

Reading: Reading Comprehension Passages, Sentence Completion (Technical and General Interest), Vocabulary and Word Analogy
Activities: Cloze tests, Logical reasoning, Advanced grammar exercises

Module:5	**Listening to Narratives**	**4 hours**

Listening: Listening to audio files of short stories, News, TV Clips/ Documentaries, Motivational Speeches in UK/ US/ global English accents.
Activity: Note-making and Interpretive exercises

Module:6	**Academic Writing and Editing**	**6 hours**

Writing: Editing/ Proofreading symbols
Citation Formats
Structure of an Abstract and Research Paper

Activity: Writing Abstracts and research paper; Work with Editing/ Proofreading exercise		
Module:7	**Team Communication**	**4 hours**
Speaking: Group Discussions and Debates on complex/ contemporary topics Discussion evaluation parameters, using logic in debates Activity: Group Discussions on general topics		
Module:8	**Career-oriented Writing**	**4 hours**
Writing: Resumes and Job Application Letters, SOP Activity: Writing resumes and SOPs		
Module:9	**Reading for Pleasure**	**4 hours**
Reading: Reading short stories Activity: Classroom discussion and note-making, critical appreciation of the short story		
Module: 10	**Creative Writing**	**4 hours**
Writing: Imaginative, narrative and descriptive prose Activity: Writing about personal experiences, unforgettable incidents, travelogues		
Module: 11	**Academic Listening**	**4 hours**
Listening: Listening in academic contexts Activity: Listening to lectures, Academic Discussions, Debates, Review Presentations, Research Talks, Project Review Meetings		
Module:12	**Reading Nature-based Narratives**	**4 hours**
Narratives on Climate Change, Nature and Environment Activity: Classroom discussions, student presentations		
Module:13	**Technical Proposals**	**4 hours**
Writing: Technical Proposals Activities: Writing a technical proposal		
Module:14	**Presentation Skills**	**4 hours**
Persuasive and Content-Specific Presentations Activity: Technical Presentations		
	Total Lecture hours:	**60 hours**

Text Book / Workbook	
1.	Oxenden, Clive and Christina Latham-Koenig. *New English File: Advanced Students Book.* Paperback. Oxford University Press, UK, 2017.
2	Rizvi, Ashraf. *Effective Technical Communication*. McGraw-Hill India, 2017.

Reference Books	
1.	Oxenden, Clive and Christina Latham-Koenig, *New English File: Advanced: Teacher's Book with Test and Assessment*. CD-ROM: Six-level General English Course for Adults. Paperback. Oxford University Press, UK, 2013.
2.	Balasubramanian, T. *English Phonetics for the Indian Students: A Workbook*. Laxmi Publications, 2016.
3.	Philip Seargeant and Bill Greenwell, *From Language to Creative Writing*. Bloomsbury Academic, 2013.
4.	Krishnaswamy, N. *Eco-English*. Bloomsbury India, 2015.
5.	Manto, Saadat Hasan. *Selected Short Stories*. Trans. Aatish Taseer. Random House India, 2012.
6.	Ghosh, Amitav. *The Hungry Tide*. Harper Collins, 2016.
7.	Ghosh, Amitav. *The Great Derangement: Climate Change and the Unthinkable*. Penguin Books, 2016.
8.	*The MLA Handbook for Writers of Research Papers*, 8th ed. 2016.

	Online Sources:
	https://americanliterature.com/short-short-stories. (75 *short* short stories) http://www.eco-ction.org/dt/thinking.html (Leopold, Aldo.―Thinking like a Mountain") https://www.esl-lab.com/; http://www.bbc.co.uk/learningenglish/; https://www.bbc.com/news; https://learningenglish.voanews.com/a/using-voa-learning-english-to-improve-listening-skills/3815547.html

Mode of evaluation: Quizzes, Presentation, Discussion, Role play, Assignments and FAT

	List of Challenging Experiments (Indicative)	
1.	Self-Introduction using SWOT	**12 hours**
2.	Writing minutes of meetings	**10 hours**
3.	Writing an abstract	**10 hours**
4.	Listening to motivational speeches and interpretation	**10 hours**
5.	Cloze Test	**6 hours**
6.	Writing a proposal	**12 hours**
	Total Laboratory Hours	**60 hours**

Mode of evaluation: Quizzes, Presentation, Discussion, Role play, Assignments and FAT

Recommended by Board of Studies	08.06.2019	
Approved by Academic Council	55	Date: 13-06-2019

ENG1903	Advanced Technical English	L	T	P	J	C
		0	0	2	4	2
Pre-requisite	Greater than 90 % EPT score	**Syllabus Version**				
		1				

Course Objectives:

1. To review literature in any form or any technical article
2. To infer content in social media and respond accordingly
3. To communicate with people across the globe overcoming trans-cultural barriers and negotiate successfully

Expected Course Outcome:

1. Analyze critically and write good reviews
2. Articulate research papers, project proposals and reports
3. Communicate effectively in a trans-cultural environment
4. Negotiate and lead teams towards success
5. Present ideas in an effective manner using web tools

Module:1	**Negotiation and Decision Making Skills through Literary Analysis**	**5 hours**

Concepts of Negotiation and Decision Making Skills

Activity: Analysis of excerpts from Shakespeare‘s –The Merchant of Venice‖ (court scene) and discussion on negotiation skills.

Critical evaluation of excerpts from Shakespeare‘s –Hamlet‖(Monologue by Hamlet) and discussion on decision making skills

Module:2	**Writing reviews and abstracts through movie interpretations**	**5 hours**

Review writing and abstract writing with competency

Activity: Watching Charles Dickens –Great Expectations‖ and writing a movie review

Watching William F. Nolan‘s –Logan‘s Run‖ and analyzing it in tune with the present scenario of depletion of resources and writing an abstract

Module:3	**Technical Writing**	**4 hours**

Stimulate effective linguistics for writing: content and style

Activity: ProofreadingStatement of Purpose

Module:4	**Trans-Cultural Communication**	**4 hours**

Nuances of Trans-cultural communication

Activity:Group discussion and case studies on trans-cultural communication.

Debate on trans-cultural communication.

Module:5	**Report Writing and Content Writing**	**4 hours**

Enhancing reportage on relevant audio-visuals

Activity: Watch a documentary on social issues and draft a report

Identify a video on any social issue and interpret

Module:6	**Drafting project proposals and article writing**	**4 hours**

Dynamics of drafting project proposals and research articles

Activity:Writing a project proposal.

Writing a research article.

Module:7	**Technical Presentations**	**4 hours**
Build smart presentation skills and strategies Activity: Technical presentations using PPT and Web tools		
	Total Lecture hours	**30 hours**
Text Book / Workbook		
1.	Raman, Meenakshi & Sangeeta Sharma. *Technical Communication: Principles and Practice*, 3rd edition, Oxford University Press, 2015.	
Reference Books		
1	Basu B.N. *Technical Writing*, 2011 Kindle edition	
2	Arathoon, Anita. *Shakespeare's The Merchant of Venice* (Text with Paraphrase), Evergreen Publishers, 2015.	
3	Kumar, Sanjay and Pushp Lata. *English Language and Communication Skills for Engineers*, Oxford University Press, India, 2018.	
4	Frantisek, Burda. *On Transcultural Communication*, 2015, LAP Lambert Academic Publishing, UK.	
5	Geever, C. Jane. *The Foundation Center's Guide to Proposal Writing*, 5th Edition, 2007, Reprint 2012 The Foundation Center, USA.	
6	Young, Milena. *Hacking Your Statement of Purpose: A Concise Guide to Writing Your SOP*, 2014 Kindle Edition.	
7	Ray, Ratri, *William Shakespeare's Hamlet*, The Atlantic Publishers, 2011.	
8	C Muralikrishna & Sunitha Mishra, *Communication Skills for Engineers*, 2nd edition, NY: Pearson, 2011.	
Mode of Evaluation: Quizzes, Presentation, Discussion, Role Play, Assignments		
List of Challenging Experiments (Indicative)		
1.	Enacting a court scene - Speaking	**6 hours**
2.	Watching a movie and writing a review	**4 hours**
3.	Trans-cultural – case studies	**2 hours**
4.	Drafting a report on any social issue	**6 hours**
5.	Technical Presentation using web tools	**6 hours**
6.	Writing a research paper	**6 hours**
J- Component Sample Projects		
1.	Short Films	
2.	Field Visits and Reporting	
3.	Case studies	
4.	Writing blogs	
5.	Vlogging	
	Total Hours (J-Component)	**60 hours**
Mode of evaluation: Quizzes, Presentation, Discussion, Role play, Assignments and FAT		
Recommended by Board of Studies	08.06.2019	
Approved by Academic Council	55	Date: 13-06-2019

FRE1001	FRANÇAIS QUOTIDIEN	L	T	P	J	C
		2	0	0	0	2
Pre-requisite	**NIL**	**Syllabus version**				
		1.0				

Course Objectives:

The course gives students the necessary background to:

- Learn the basics of French language and to communicate effectively in French in their day to day life.
- Achieve functional proficiency in listening, speaking, reading and writing
- Recognize culture-specific perspectives and values embedded in French language.

Expected Course Outcome:

The students will be able to :

- Identify in French language the daily life communicative situations via personal pronouns, emphatic pronouns, salutations, negations and interrogations.
- Communicate effectively in French language via regular / irregular verbs.
- Demonstrate comprehension of the spoken / written language in translating simple sentences.
- Understand and demonstrate the comprehension of some particular new range of unseen written materials
- Demonstrate a clear understanding of the French culture through the language studied

Module: 1	**Expressions simples**	**3 hours**

Les Salutations, Les nombres (1-100), Les jours de la semaine, Les mois de l'année, Les Pronoms Sujets, Les Pronoms Toniques, La conjugaison des verbes irréguliers- avoir / être / aller / venir / faire etc.

Savoir-faire pour:Saluer, Se présenter, Présenter quelqu'un, Etablir des contacts

Module: 2	**La conjugaison des verbes réguliers**	**3 hours**

La conjugaison des verbes réguliers, La conjugaison des verbes pronominaux, La Négation, L'interrogation avec _Est-ce que ou sans Est-ce que'.

Savoir-faire pour:

Chercher un(e) correspondant(e), Demander des nouvelles d'une personne.

Module: 3	**La Nationalité du Pays, L'article (défini/ indéfini), Les prépositions**	**6 hours**

La Nationalité du Pays, L'article (défini/ indéfini), Les prépositions (à/en/au/aux/sur/dans/avec etc.), L'article contracté, Les heures en français, L'adjectif (La Couleur, L'adjectif possessif, L'adjectif démonstratif/ L'adjectif interrogatif (quel/quelles/quelle/quelles), L'accord des adjectifs avec le nom, L'interrogation avec Comment/ Combien / Où etc.

Savoir-faire pour:

Poser des questions, Dire la date et les heures en français,

Module: 4	**La traduction simple**	**4 hours**

La traduction simple :(français-anglais / anglais –français),

Savoir-faire pour :

Faire des achats, Comprendre un texte court, Demander et indiquer le chemin.

Module: 5	**L'article Partitif, Mettez les phrases aux pluriels**	**5 hours**

L'article Partitif, Mettez les phrases aux pluriels, Faites une phrase avec les mots donnés, Trouvez les questions.

Savoir-faire pour :

Répondez aux questions générales en français, Exprimez les phrases données au Masculin ou au Féminin, Associez les phrases.

Module: 6	**Décrivez :**	**3 hours**
Décrivez: La Famille / La Maison / L'université / Les Loisirs / La Vie quotidienne etc.		
Module: 7	Dialogue	**4 hours**
Dialogue: 1. Décrire une personne. 2. Des conversations à la cafeteria. 3. Des conversations avec les membres de la famille 4. Des dialogues entre les amis.		
Module: 8	**Contemporary issues**	**2 hours**
Guest lectures / Natives speakers		
Total Lecture hours		**30 hours**

Text Book(s)	
1.	Fréquence jeunes-1, Méthode de français, G. Capelle et N.Gidon, Hachette, Paris, 2010.
2.	Fréquence jeunes-1, Cahier d'exercices, G. Capelle et N.Gidon, Hachette, Paris, 2010.
Reference Books	
1.	CONNEXIONS 1, Méthode de français, Régine Mérieux, Yves Loiseau,Les Éditions Didier, 2010.
2.	CONNEXIONS 1, Le cahier d'exercices, Régine Mérieux, Yves Loiseau, Les Éditions Didier, 2010
3.	ALTER EGO 1, Méthode de français, Annie Berthet, Catherine Hugo, Véronique M. Kizirian, Béatrix Sampsonis, Monique Waendendries, Hachette livre Paris 2011
4.	ALTER EGO 1, Le cahier d'activités, Annie Berthet, Catherine Hugo, Béatrix Sampsonis, Monique Waendendries, Hachette livre, Paris 2011

Mode of Evaluation: CAT / Assignment / Quiz / Seminar / FAT			
Recommended by Board of Studies	26.02.2016		
Approved by Academic Council	41st ACM	**Date**	17.06.2016

FRE2001	FRANÇAIS PROGRESSIF	L	T	P	J	C
		2	0	2	0	3
Pre-requisite	**Français Quotidien**	**Syllabus version**				
		1.0				

Course Objectives:

The course gives students the necessary background to:

1. Understand isolated sentences and frequently used expressions in relation to immediate priority areas (personal or family information, shopping, close environment, work).
2. Communicate in simple and routine tasks requiring only a simple and direct exchange of information on familiar and habitual topics.
3. Enable students to describe with simply means his training, his immediate environment and evoke familiar and habitual subjects, evoke subjects that correspond to immediate needs.

Expected Course Outcome:

The students will be able to :

1. Understand expressions in French.
2. Create senteces by using frequent lexicon related to himself, his family, his close environment (family, shopping, work, school, etc).
3. Understand simple, clear messages on internet, authentic documents.
4. Analyse predictable information in common documents, such as advertisements, flyers, menus, schedules, simple personal letters.
5. Create simple and routine tasks.
6. Create simple and direct exchange of information on familiar activities and topics.

Module: 1	**Expressions simples**	**8 hours**

La vie quotidiennes - Le verbe pronominal - Le passé composé avec l'auxiliaire - avoir et être- le passé récent : venir de + infinitif - Le comparatif - Le superlatif - Les mots interrogatifs (les trois formes)

Savoir-faire pour : Faire des achats, faire des commandes dans un restaurant, poser des questions.

Module: 2	**Les activitiés quotidiennes**	**6 hours**

La vie privée et publique (Les achats, Les voyages, les transports-La nourriture, etc.) - Les lieux de la ville - Les mots du savoir-vivre - Les pronoms indéfinis - Les pronoms démonstratifs - Les pronoms compléments objets directs/ indirects - La formation du future simple et future proche

Savoir-faire pour : Réserver les billets pour le voyage, réserver les chambres dans un hôtel, S'informer sur les lieux de la ville, indiquer la direction à un étranger.

Module: 3	**Les activités de loisirs**	**7 hours**

Les loisirs (sports/spectacles/activités) - Les moments de la journée, de l'année- La fête indienne et française – Les goûts - L'impératif - La négation de l'impératif-La place du pronom à l'impératif avec un verbe pronominal.

Savoir-faire pour : Parler de ses goûts, raconter les vacances, formuler des phrases plus compliquées, Raconter les souvenirs de l'enfance, parler sur la tradition de son pays natal.

Module: 4	**La Francophonie**	**7 hours**

L'espace francophone - Première approche de la société française – La consommation alimentaire – caractériser un objet – décrire une tenue - Le pronom relatif (qui/que/dont/où)

Savoir-faire pour :

Articles de la presse-Portrait d'une personne-Cartes et messages d'invitation, d'acceptation ou de refus -Article de presse - rédaction d'un événement.

Module: 5	**La culture française**	**5 hours**

Parler de ses activités quotidiennes - les fêtes en France – Parler de sa famille – réserver un billet à

l‘agence - la gastronomie française				
Module: 6	**La description**			**5 hours**
Décrire physiquement une personne – les vacances – les achats – réserver une chambre dans un hôtel – les plus grands français - raconter des évènements passés				
Module: 7	**S’exprimer**			**5 hours**
Parler du climat - parcours francophone – placer une commande au restaurant –- la mode - parler de son projet d‘avenir.				
Module: 8	**Contemporary issues**			**2 hours**
Guest lectures / Natives speakers				
Total Lecture hours				**45 hours**
Text Book(s)				
1.	Alter Ego 1, Méthode de français, Annie Berthet, Hachette, Paris 2010.			
2.	Alter Ego 1, Cahier d‘exercices, Annie Berthet, Hachette, Paris 2010.			
Reference Books				
1.	CONNEXIONS 1, Méthode de français, Régine Mérieux, Yves Loiseau,Les Éditions Didier, 2010.			
2.	CONNEXIONS 1, Le cahier d‘exercices, Régine Mérieux, Yves Loiseau, Les Éditions Didier, 2010			
3.	Fréquence jeunes-1, Méthode de français, G. Capelle et N.Gidon, Hachette, Paris, 2010.			
Mode of Evaluation: CAT / Assignment / Quiz / Project / Seminar / FAT				
Recommended by Board of Studies	26.02.2016			
Approved by Academic Council	41st ACM	**Date**	17.06.2016	

GER1001	GRUNDSTUFE DEUTSCH	L	T	P	J	C
		2	0	0	0	2
Pre-requisite	**Nil**	**Syllabus version**				
		1.0				

Course Objectives:

The course gives students the necessary background to:

1. Demonstrate Proficiency in reading, writing, and speaking in basic German. Learning vocabulary related to profession, education centres, day-to-day activities, food, culture, sports and hobby, family set up, workplace, market and classroom activities are essential.
2. Make the students industry oriented and make them adapt in the German culture.

Expected Course Outcome:

The students will be able to

1. Remember greeting people, introducing oneself and understanding basic expressions in German.
2. Understand basic grammar skills to use these in a meaning way.
3. Remember beginner's level vocabulary
4. Create sentences in German on a variety of topics with significant precision and in detail.
5. Apply good comprehension of written discourse in areas of special interests.

Module: 1		**3 hours**

Begrüssung, Landeskunde, Alphabet, Personalpronomen, Verben- heissen, kommen, wohnen, lernen, Zahlen (1-100), W-Fragen, Aussagesätze, Nomen- Singular und Plural, der Artikel -Bestimmter- Unbestimmter Artikel)

Lernziel :

Sich vorstellen, Grundlegendes Verständnis von Deutsch, Deutschland in Europa

Module: 2		**3 hours**

Konjugation der Verben (regelmässig /unregelmässig),das Jahr- Monate, Jahreszeiten und die Woche, Hobbys, Berufe, Artikel, Zahlen (Hundert bis eine Million), Ja-/Nein- Frage, Imperativ mit „Sie‘‘

Lernziel:

Sätze schreiben, über Hobbys, Berufe erzählen, usw

Module: 3		**5 hours**

Possessivpronomen, Negation, Kasus (Bestimmter- Unbestimmter Artikel) Trennbareverben, Modalverben, Uhrzeit, Präpositionen, Lebensmittel, Getränkeund Essen, Farben, Tiere

Lernziel :

Sätze mit Modalverben, Verwendung von Artikel, Adjektiv beim Verb

Module: 4		**5 hours**

Übersetzung: (Deutsch – Englisch / Englisch – Deutsch)

Lernziel :

Die Übung von Grammatik und Wortschatz

Module: 5		**5 hours**

Leserverständnis. Mindmap machen, Korrespondenz- Briefe und Email

Lernziel:

Übung der Sprache, Wortschatzbildung

Module: 6		**3 hours**

Aufsätze : Die Familie, Bundesländer in Deutschland, Ein Fest in Deutschland,

Lernziel : Aktiver, selbständiger Gebrauch der Sprache			
Module: 7			**4 hours**
Dialoge: a) Gespräche mit einem/einer Freund /Freundin. b) Gespräche beim Einkaufen ; in einem Supermarkt ; in einer Buchhandlung ; c) in einem Hotel - an der Rezeption ; ein Termin beim Arzt. d) Ein Telefongespräch ; Einladung–Abendessen			
Module: 8	**Contemporary issues**		**2 hours**
Guest Lectures / Native Speakers Einleitung in die deustche Kultur und Politik			
Total Lecture hours			**30 hours**
Text Book(s)			
1.	Netzwerk Deutsch als Fremdsprache A1, Stefanie Dengler, Paul Rusch, Helen Schmtiz, Tanja Sieber, Klett-Langenscheidt Verlag, München : 2013		
Reference Books			
1.	Lagune, Hartmut Aufderstrasse, Jutta Müller, Thomas Storz, 2012.		
2.	Deutsche Sprachlehre für Ausländer, Heinz Griesbach, Dora Schulz, 2013		
3.	Studio d A1, Hermann Funk, Christina Kuhn, CorneslenVerlag, Berlin: 2010		
4.	Tangram Aktuell-I, Maria-Rosa, SchoenherrTil, Max Hueber Verlag, Muenchen: 2012		
	www.goethe.de wirtschaftsdeutsch.de hueber.de klett-sprachen.de www.deutschtraning.org		
Mode of Evaluation: CAT / Assignment / Quiz / Seminar / FAT			
Recommended by Board of Studies	04.03.2016		
Approved by Academic Council	41st ACM	**Date**	17.06.2016

GER2001	MITTELSTUFE DEUTSCH	L	T	P	J	C
		2	0	2	0	3
Pre-requisite	**Grundstufe Deutsch**	**Syllabus version**				
		1.0				

Course Objectives:

The course gives students the necessary background to:

1. Improve the communication skills in German language
2. Improve the listening and understanding capability of German FM Radio, and TV Programmes, Films
3. Build the confidence of the usage of German language and better understanding of the culture

Expected Course Outcome:

The students will be able to

1. Create proficiency in advanced grammar and rules
2. Understand the texts including scientific subjects.
3. Create the ability of listening and speaking in real time situations.
4. Create the vocabulary in different context-based situations.
5. Create written communication in profession life, like replying or sending E-mails and letters in a company.
6. Create communication related to simple and routine tasks.

Module: 1	**Proficiency in Advanced Grammar**	**8 hours**

Grammatik : Tempus- Perfekt, Präteritum, Plusquamperfekt, Futur-I, Futur-II, Wiederholung der Grundstufen grammatik
Lernziel: Sätzeschreiben in verschiedenen Zeiten.

Module: 2	**Understanding of Technical Texts**	**6 hours**

Grammatik : Passiv, Personalpronomen (Nominativ, Akkusativ, Dativ)
Lernziel: Passiv, Formen des Personalpronomens

Module: 3	**Understanding of Scientific texts**	**7 hours**

Adjektivdeklination, Nebensatz, Präpositionen mit Akkusativ und Dativ, Infinitiv Sätze
Lernziel: Verbindung zwischen Adjektiv beim Nomen

Module: 4	**Communicating in Real Time Situations**	**7 hours**

Übersetzung:Technische Terminologie, wissenschaftliche, literarische Texte aus dem Deutschen ins Englische und umgekehrt,
Lernziel : Übung von Grammatik und Wortschatz

Module: 5	**Acquisition of the Vocabulary of the advanced Level**	**5 hours**

Hörverständnis durch Audioübung :Familie, Leben in Deutschland, Am Bahnhof,
Videos : Politik, Historie, Tagesablauf in eineranderen Stadt,
Lernziel : Übung der Sprache

Module: 6	**Ability to Communicate in Professional Life**	**5 hours**

Hörverständnis durch Audioübung: Überberühmte Persönlichkeiten, Feste in Deutschland,
Videos:Wetter, An der Universität,ein Zimmer buchen, Studentenleben,Städteund Landeskunde
Lernziel: Hörverständnis, Landeskunde

Module: 7	**Ability to Communicate in Task-based Situations**	**5 hours**

Hörverständnis durch Audioübung: FM Radio aus Deutschland
Videos: Fernseher aus Deutschland
Lernziel: LSRW Fähigkeiten

Module: 8	**Contemporary issues**	**2 hours**

Total Lecture hours				**45 hours**
Text Book(s)				
1.	Text Book: 1. TangramAktuell II, Rosa Maria Dallapizza, Beate Blüggel, Max Hueber Verlag, München : 2010			
Reference Books				
1.	Themen Aktuell, Heiko Bock, Mueller Jutta, Max Hueber Verla, Muenchen : 2010			
2.	Deutsch Sprachlehre fuer Auslaender, Schulz Griesbach, Max Hueber Verlag, Muenchen : 2012			
3.	Lagune, Deutsch als Fremdsprache, Jutta Müller, Storz Thomas, Hueber Verlag, Ismaning : 2013			
4.	Studio d A1, Hermann Funk, Christina Kuhn, Max HuerberVerlag, München : 2011			
Mode of Evaluation: CAT / Assignment / Quiz / Seminar / FAT				
Recommended by Board of Studies	04.03.2016			
Approved by Academic Council	41st ACM	**Date**	17.06.2016	

JAP1001	JAPANESE FOR BEGINNERS	L	T	P	J	C
		2	0	0	0	2
Pre-requisite	Nil	Syllabus version				
		1.0				

Course Objectives:

The course gives students the necessary background to:

1. Develop four basic skills related to reading, listening, speaking and writing Japanese language.
2. Instill in learners an interest in Japanese language by teaching them culture and general etiquettes.
3. Recognize, read and write Hiragana and Katakana.

Expected Course Outcomes:

Students will be able to:

1. Remember Japanese alphabets and greet in Japanese.
2. Understand pronouns, verbs form, adjectives and conjunctions in Japanese.
3. Remember time and dates related vocabularies and express them in Japanese.
4. Create simple questions and its answers in Japanese.
5. Understand the Japanese culture and etiquettes.

Module: 1	Introduction to Japanese syllables and Greetings	4 hours

Introduction of Japanese language, alphabets; Hiragana, katakana, and Kanji Pronunciation, vowels and consonants.

Hiragana – writing and reading; Vocabulary: 50 Nouns and 20 pronouns, Greetings.

Module: 2	Demonstrative Pronouns	4 hours

Grammar: N1 wa N2 desu, Japanese Numerals, Demonstrative pronoun - Kore, Sore, Are and Dore

(This, That, Over there, which) Kono, sono, Ano and Dono (this, that, over there, which) Kochira, Sochira, Achira and

Dochira. this way...) Koko, Soko, Asoko and Doko (Here, There…. location)

Module: 3	Verbs and Sentence formation	4 hours

Classification of verbs Be verb desu Present and Present negative Basic structure of sentence (Subject+ Object+

Verb) Katakana-reading and writing

Module: 4	Conjunction and Adjectives	4 hours

Conjunction-Ya…..nado Classification of Adjectives _I‘ and _na‘-ending Set phrase – Onegaishimasu – Sumimasen,

wakarimasen Particle –Wa, Particle-Ni _Ga imasu‘ and _Ga arimasu‘ for Existence of living things and non-living things

Particle- Ka, Ni, Ga

Module: 5	Vocabulary and its Meaning	4 hours

Days/ Months /Year/Week (Current, Previous, Next, Next to Next) ; Nation, People and Language Relationship of

family (look and learn); Simple kanji recognition

Module: 6	Forming questions and giving answers	4 hours

Classification of Question words (Dare, Nani, Itsu, Doyatte, dooshite, Ikutsu, Ikura); Classification of Te forms, Polite

form of verbs

Module: 7	Expressing time, position and directions	4 hours

Classification of question words (Doko, Dore, Dono, Dochira); Time expressions (Jikan), Number of

hours, Number of months, calendar of a month; Visit the departmental store, railway stations, Hospital (Byoki), office and University			
Module: 8	**Contemporary issues**		**2 hours**
Total Lecture hours			**30 hours**
Text Book(s):			
1.	The Japan Foundation (2017), Marugoto Japanese Language and Culture Starter A1 Coursebook For Communicative Language Competences, New Delhi: Goyal Publishers (9788183078047)		
2.	Banno, Eri et al (2011), Genki: An Integrated Course in Elementary Japanese I [Second Edition], Japan: The Japan Times.		
Reference Book(s):			
1.	Japanese for Busy people (2011) video CD, AJALT, Japan.		
2.	Carol and Nobuo Akiyama (2010), The Fast and Fun Way, New Delhi: Barron's Publication		
Mode of Evaluation: CAT , Quiz and Digital Assignments			
Recommended by Board of Studies	24.10.2018		
Approved by Academic Council	53rd ACM	**Date**	13.12.2018

ESP1001	ESPAÑOL FUNDAMENTAL	L	T	P	J	C
		2	0	0	0	2
Pre-requisite	**Nil**	**Syllabus version**				
		1.0				

Course Objectives:

The course gives students the necessary background to:

- Demonstrate Proficiency in reading, writing, and speaking in basic Spanish. Learning vocabulary related to profession, education centres, day today activities, food, culture, sports and hobby, family set up, workplace, market and classroom activities is essential.
- Demonstrate the ability to describe things and will be able to translate into English and vice versa.
- Describe in simple terms (both in written and oral form) aspects of their background, immediate environment and matters in areas of immediate need.

Expected Course Outcome:

The students will be able to

- Remember greetings, giving personal details and Identify genders by using correct articles
- Apply the correct use of SER, ESTAR and TENER verb for describing people, place and things
- Create opinion about time and weather conditions by knowing months, days and seasons in Spanish
- Create opinion about people and places by using regular verbs
- Apply reflexive verbs for writing about daily routine and create small paragraphs about hometown, best friend and family

Module: 1	Abecedario, Saludos y Datos personales: Origen, Nacionalidad, Profesión	**3 hours**
Competencia Gramática: Vocales y Consonantes. Artículos definidos e indefinidos (Numero y Genero). Competencia Escrita: Saludos y Datos personales		
Module: 2	Edad y posesión. Números (1-20)	**3 hours**
Competencia Gramática: Pronombres personales. Adjetivos. Los verbos SER y TENER. Competencia Escrita: Escribe sobre mismo/a y los compañeros de la clase		
Module: 3	Vocabulario de Mi habitación. Colores. Descripción de lugares y cosas	**5 hours**
Competencia Gramática: Adjetivos posesivos. El uso del verbo ESTAR. Diferencia entre SER y ESTAR. Competencia Escrita: Mi habitación		
Module: 4	Mi familia. Números (21-100). Direcciones.Expresar la hora. Los meses del año.	**5hours**
Competencia Gramática: Frases preposicionales. Uso del HAY. La diferencia entre MUY y MUCHO. Uso del verbo GUSTAR Competencia Escrita: Mi familia. Dar opiniones sobre tiempo		
Module: 5	Expresar fechas y el tiempo. Dar opiniones sobre personas y lugares.	**5 hours**
Competencia Gramática: Los verbos regulares (-AR, -ER, -IR) en el presente. Adjetivos demostrativos. Competencia Escrita: Mi mejor amigo/a. Expresar fechas. Traducción ingles a español y Español a Ingles.		
Module: 6	Describir el diario. Las actividades cotidianas.	**3 hours**

Competencia Gramática: Los Verbos y pronombres reflexivos. Los verbos pronominales con e/ie, o/ue, e/i, u/ue. Competencia Escrita: El horario. Traducción ingles a español y Español a Ingles.		
Module: 7	Dar opiniones sobre comidas y bebidas. Decir lo que está haciendo. Describir mi ciudad y Ubicar los sitios en la ciudad.	**4 hours**
Competencia Gramática: Los verbos irregulares. Estar + gerundio. Poder + Infinitivo. Competencia Escrita: Conversación en un restaurante. Traducción ingles a español y Español a Ingles.Mi ciudad natal. Mi Universidad. La clase.Mi fiesta favorita.		
Module: 8	**Contemporary issues**	**2 hours**
Total Lecture hours		**30 hours**

Text Book(s)	
1.	Text Book: ―Aula Internacional 1‖, Jaime Corpas, Eva Garcia, Agustin Garmendia, Carmen Soriano Goyal Publication; reprinted Edition, (2010)
Reference Books	
1.	―¡Acción Gramática!‖ Phil Turk and Mike Zollo, Hodder Murray, London 2006. ―Practice makes perfect: Spanish Vocabulary‖, Dorothy Richmond, McGraw Hill Contemporary, USA,2012.
2.	―Practice makes perfect: Basic Spanish‖, Dorothy Richmond, McGraw Hill Contemporary, USA 2009.
3.	―Pasaporte A1 Foundation‖, Matilde Cerrolaza Aragón, Óscar Cerrolaza Gili, Begoña Llovet Barquero, Edelsa Grupo, España, 2010.

Recommended by Board of Studies	22.02.2016		
Approved by Academic Council	41st ACM	**Date**	17.06.2016

ESP2001	ESPAÑOL INTERMEDIO	L	T	P	J	C
		2	0	2	0	3
Pre-requisite		Syllabus version				
		1.0				

Course Objectives:

The course gives students the necessary background to:

1. Enable students to read, listen and communicate in Spanish in their day to day life.
2. Enable students to describe situations by using present, past and future tenses in Spanish.
3. Enable to develop the comprehension skill in Spanish language.

Expected Course Outcome:

The students will be able to

1. Create sentences in near future and future tenses and correctly using the prepositions like POR and PARA
2. Create sentences in preterito perfecto and correctly use the direct and indirect object pronouns
3. Create sentences related to likes and dislikes and also give commands in formal and informal way
4. Create sentences in past tense by using imperfect and idefinido forms and describe past events
5. Create conversations in Spanish at places like restaurants, hotels, Shops and Railway stations
6. Understand about different Spanish speaking countries and its culture and traditions.

Module: 1	Números (101 – 1 millón). Expresar los planes futuros. Los númerosordinales.	7 hours

Competencia Gramática: Futuros cercanos (Ir+a+Infinitivo). Futuros (Verbos regulares e irregulares).Uso del POR y PARA.
Competencia Escrita: Traducción ingles a español y español a Ingles.
Comprensión - Los textos y Videos

Module: 2	Las ropas, colores y tamaños. Costar, valer, descuentos y rebajas	8 hours

Competencia Gramática: Pronombres objetivos directos e indirectos. El verbo Gustar y Disgustar.
Competencia Escrita: Traducción ingles a español y español a Ingles. Comprensión - Los textos y Videos

Module: 3	Escribir un Correo electrónico formal e informal.	7 hours

Competencia Gramática: Imperativos formales e informales. Pretérito perfecto.
Competencia Escrita: Traducción ingles a español y español a Ingles.
Comprensión - Los textos y Videos

Module: 4	Currículo Vitae. Presentarse en una entrevista informal.	6 hours

Competencia Gramática: Pretérito imperfecto. Pretérito indefinido.
Competencia Escrita: Traducción ingles a español y español a Ingles.
Comprensión - Los textos y Videos

Module: 5	Introducción personal, Expresar los planes futuros.	5 hours

Comprensión oral: Introducción personal, Expresar los planes futuros. ¿Qué vas a hacer en las próximas vacaciones?
Comprensión auditiva: Las preguntas sobre un cuento auditivo. Relacionar el audio con las imágenes. Las preguntas basadas en canciones.
Medio de transporte: Comprar y Reservar billetes.

Module: 6	Diálogos entre dos	5 hours

Comprensión oral: Diálogos entre dos (cliente y tendero de ropas, pasajero y empleado, en un restaurante, Reservación de habitación en un hotel). Presentación en una entrevista.
Comprensión auditiva: Las preguntas basadas en canciones. Las preguntas basadas en diálogos.

Module: 7	**Presentación de los países hispánicos.**	**5 hours**
Comprensión oral: Dialogo entre un médico y paciente. Presentación de los países hispánicos. Describir su infancia. Describir vacaciones últimas o las actividades de último fin de semana. Comprensión auditiva: Rellenar los blancos del cuento en pasado. Las preguntas basadas en el cuento. Las preguntas basadas en un anuncio		
Module: 8	**Contemporary issues**	**2 hours**
Total Lecture hours		**45 hours**

Text Book(s)	
1.	−Aula Internacional 1‖, Jaime Corpas, Eva Garcia, Agustin Garmendia, Carmen Soriano Goyal Publication; reprinted Edition, Delhi (2010)
Reference Books	
1.	−¡Acción Gramática!‖ Phil Turk and Mike Zollo, Hodder Murray, London 2006.
2.	−Practice makes perfect: Spanish Vocabulary‖, Dorothy Richmond, McGraw Hill Contemporary, USA,2012.
3.	−Practice makes perfect: Basic Spanish‖, Dorothy Richmond, McGraw Hill Contemporary, USA 2009.
4.	−Pasaporte A1 Foundation‖, Matilde Cerrolaza Aragón, Óscar Cerrolaza Gili, Begoña Llovet Barquero, Edelsa Grupo, España, 2010.
	Authors, book title, year of publication, edition number, press, place

Recommended by Board of Studies	22-02-2016		
Approved by Academic Council	41st ACM	**Date**	17-06-2016

HUM1021	ETHICS AND VALUES	L	T	P	J	C
		2	0	0	0	2
Pre-requisite	Nil	Syllabus version				
		1.2				

Course Objectives:

1. To understand and appreciate the ethical issues faced by an individual in profession, society and polity
2. To understand the negative health impacts of certain unhealthy behaviors
3. To appreciate the need and importance of physical, emotional health and social health

Expected Course Outcome:

Students will be able to:

1. Follow sound morals and ethical values scrupulously to prove as good citizens
2. Understand varioussocial problems and learn to act ethically
3. Understand the concept of addiction and how it will affect the physical and mental health
4. Identify ethical concerns in research and intellectual contexts, including academic integrity, use and citation of sources, the objective presentation of data, and the treatment of human subjects
5. Identify the main typologies, characteristics, activities, actors and forms of cybercrime

Module: 1	Being good and responsible	5 hours

Gandhian values such as truth and non-violence – comparative analysis on leaders of past and present – society‘s interests versus self-interests–Personal Social Responsibility: Helping the needy, charity and serving the society.

Module: 2	Social Issues 1	4 hours

Harassment – types - Prevention of harassment, violence and terrorism

Module: 3	Social Issues 2	4 hours

Corruption: ethical values, causes, impact, laws, prevention – electoral malpractices white collar crimes – tax evasions – unfair trade practices

Module: 4	Addiction and Health	3 hours

Peer pressure - Alcoholism: ethical values, causes, impact, laws, prevention – Ill effects of smoking – Prevention of Suicides

Sexual Health: Prevention and impact of pre-marital pregnancy and Sexually Transmitted Diseases

Module: 5	Drug Abuse	4 hours

Abuse of different types of legal and illegal drugs: ethical values, causes, impact, laws and prevention

Module: 6	Personal and Professional Ethics	3 hours

Dishonesty - Stealing - Malpractices in Examinations – Plagiarism

Module: 7	Abuse of technologies	4 hours

Hacking and other cyber crimes, addiction to mobile phone usage, video games and social networking websites

Module: 8	Contemporary issues	3 hours
	Total Lecture hours	30 hours

Reference Books

1. Dhaliwal, K.K (2016), ‒Gandhian Philosophy of Ethics: A Study of Relationship between his Presupposition and Precepts, Writers Choice, New Delhi, India
2. Vittal, N (2012), ‒Ending Corruption? - How to Clean up India?‖, Penguin Publishers, UK
3. Pagliaro, L.A. and Pagliaro, A.M (2012), ‒Handbook of Child and Adolescent Drug and Substance Abuse: Pharmacological , Developmental and Clinical Considerations‖, Wiley Publishers, U.S.A

4.	Pandey, P. K (2012), –Sexual Harassment and Law in India‖, Lambert Publishers, Germany		
Mode of Evaluation: CAT, Assignment, Quiz, FAT and Seminar			
Recommended by Board of Studies	26.07.2017		
Approved by Academic Council	46th ACM	**Date**	24.08.2017

MAT1011	Calculus for Engineers	L	T	P	J	C
		3	0	2	0	4
Pre-requisite	**10+2 Mathematics or MAT1001**	**Syllabus Version**				
		1.0				

Course Objectives :

1. To provide the requisite and relevant background necessary to understand the other important engineering mathematics courses offered for Engineers and Scientists.
2. To introduce important topics of applied mathematics, namely Single and Multivariable Calculus and Vector Calculus etc.
3. To impart the knowledge of Laplace transform, an important transform technique for Engineers which requires knowledge of integration

Expected Course Outcomes:

At the end of this course the students should be able to

1. apply single variable differentiation and integration to solve applied problems in engineering and find the maxima and minima of functions
2. understand basic concepts of Laplace Transforms and solve problems with periodic functions, step functions, impulse functions and convolution
3. evaluate partial derivatives, limits, total differentials, Jacobians, Taylor series and optimization problems involving several variables with or without constraints
4. evaluate multiple integrals in Cartesian, Polar, Cylindrical and Spherical coordinates.
5. understand gradient, directional derivatives, divergence, curl and Greens‘, Stokes,Gauss theorems
6. demonstrate MATLAB code for challenging problems in engineering

Module:1	**Application of Single Variable Calculus**	**9 hours**

Differentiation- Extrema on an Interval-Rolle‘s Theorem and the Mean Value Theorem-Increasing and Decreasing functions and First derivative test-Second derivative test-Maxima and Minima-Concavity. Integration-Average function value - Area between curves - Volumes of solids of revolution - Beta and Gamma functions–interrelation

Module:2	**Laplace transforms**	**7 hours**

Definition of Laplace transform-Properties-Laplace transform of periodic functions-Laplace transform of unit step function, Impulse function-Inverse Laplace transform-Convolution.

Module:3	**Multivariable Calculus**	**4 hours**

Functions of two variables-limits and continuity-partial derivatives –total differential-Jacobian and its properties.

Module:4	**Application of Multivariable Calculus**	**5 hours**

Taylor‘s expansion for two variables–maxima and minima–constrained maxima and minima-Lagrange‘s multiplier method.

Module:5	**Multiple integrals**	**8 hours**
Evaluation of double integrals–change of order of integration–change of variables between Cartesian and polar co-ordinates - Evaluation of triple integrals-change of variables between Cartesian and cylindrical and spherical co-ordinates- evaluation of multiple integrals using gamma and beta functions.		
Module:6	**Vector Differentiation**	**5 hours**
Scalar and vector valued functions – gradient, tangent plane–directional derivative-divergence and curl–scalar and vector potentials–Statement of vector identities-Simple problems		
Module:7	**Vector Integration**	**5 hours**
line, surface and volume integrals - Statement of Green's, Stoke's and Gauss divergence theorems -verification and evaluation of vector integrals using them.		
Module:8	**Contemporary Issues**	**2 hours**
Industry Expert Lecture		
	Total Lecture hours:	**45 hours**

Text Book(s)

[1] Thomas' Calculus, George B.Thomas, D.Weir and J. Hass, 13th edition, Pearson, 2014.
[2] Advanced Engineering Mathematics, Erwin Kreyszig, 10th Edition, Wiley India, 2015.

Reference Books

1. Higher Engineering Mathematics, B.S. Grewal, 43rd Edition ,Khanna Publishers, 2015
2. Higher Engineering Mathematics, John Bird, 6th Edition, Elsevier Limited, 2017.
3. Calculus: Early Transcendentals, James Stewart, 8th edition, Cengage Learning, 2017.
4. Engineering Mathematics, K.A.Stroud and Dexter J. Booth, 7th Edition, Palgrave Macmillan (2013)

Mode of Evaluation

Digital Assignments, Quiz, Continuous Assessments, Final Assessment Test

List of Challenging Experiments (Indicative)		
1.	Introduction to MATLAB through matrices, and general Syntax	3 hours
2	Plotting and visualizing curves and surfaces in MATLAB – Symbolic computations using MATLAB	3 hours
3.	Evaluating Extremum of a single variable function	3 hours
4.	Understanding integration as Area under the curve	3 hours
5.	Evaluation of Volume by Integrals (Solids of Revolution)	3 hours
6.	Evaluating maxima and minima of functions of several variables	3 hours
7.	Applying Lagrange multiplier optimization method	2 hours
8.	Evaluating Volume under surfaces	2 hours
9.	Evaluating triple integrals	2 hours
10.	Evaluating gradient, curl and divergence	2 hours
11.	Evaluating line integrals in vectors	2 hours

12.	Applying Green's theorem to real world problems		2 hours
	Total Laboratory Hours		**30 hours**
Mode of Assessment:			
Weekly assessment, Final Assessment Test			
Recommended by Board of Studies	12-06-2015		
Approved by Academic Council	No. 37	Date	16-06-2015

MAT2001	Statistics for Engineers	L	T	P	J	C
		3	0	2	0	4
Prerequisites	**MAT1011 – Calculus for Engineers**	**Syllabus Version**: 1.1				

Course Objectives :

1. To provide students with a framework that will help them choose the appropriate descriptive methods in various data analysis situations.
2. To analyse distributions and relationship of real-time data.
3. To apply estimation and testing methods to make inference and modelling techniques for decision making.

Expected Course Outcome:

At the end of the course the student should be able to:

1. Compute and interpret descriptive statistics using numerical and graphical techniques.
2. Understand the basic concepts of random variables and find an appropriate distribution for analysing data specific to an experiment.
3. Apply statistical methods like correlation, regression analysis in analysing, interpreting experimental data.
4. Make appropriate decisions using statistical inference that is the central to experimental research.
5. Use statistical methodology and tools in reliability engineering problems.
6. demonstrate R programming for statistical data

Module: 1	Introduction to Statistics	6 hours

Introduction to statistics and data analysis-Measures of central tendency –Measures of variability-[Moments-Skewness-Kurtosis (Concepts only)].

Module: 2	Random variables	8 hours

Introduction -random variables-Probability mass Function, distribution and density functions - joint Probability distribution and joint density functions- Marginal, conditional distribution and density functions- Mathematical expectation, and its properties Covariance , moment generating function – characteristic function.

Module: 3	Correlation and regression	4 hours

Correlation and Regression – Rank Correlation- Partial and Multiple correlation- Multiple regression.

Module: 4	Probability Distributions	7 hours

Binomial and Poisson distributions – Normal distribution – Gamma distribution – Exponential distribution – Weibull distribution.

Module: 5	Hypothesis Testing I	4 hours
Testing of hypothesis – Introduction-Types of errors, critical region, procedure of testing hypothesis-Large sample tests- Z test for Single Proportion, Difference of Proportion, mean and difference of means.		
Module: 6	**Hypothesis Testing II**	**9 hours**
Small sample tests- Student‘s t-test, F-test- chi-square test- goodness of fit - independence of attributes- Design of Experiments - Analysis of variance – one and two way classifications - CRD-RBD- LSD.		
Module: 7	**Reliability**	**5 hours**
Basic concepts- Hazard function-Reliabilities of series and parallel systems- System Reliability - Maintainability-Preventive and repair maintenance- Availability.		
Module: 8	**Contemporary Issues**	**2 hours**
Industry Expert Lecture		
	Total Lecture hours	**45 hours**

Text book(s)

- Probability and Statistics for engineers and scientists, R.E.Walpole, R.H.Myers, S.L.Mayers and K.Ye, 9th Edition, Pearson Education (2012).
- Applied Statistics and Probability for Engineers, Douglas C. Montgomery, George C. Runger, 6th Edition, John Wiley & Sons (2016).

Reference books

- Reliability Engineering, E.Balagurusamy, Tata McGraw Hill, Tenth reprint 2017.
- Probability and Statistics, J.L.Devore, 8th Edition, Brooks/Cole, Cengage Learning (2012).
- Probability and Statistics for Engineers, R.A.Johnson, Miller Freund‘s, 8th edition, Prentice Hall India (2011).
- Probability, Statistics and Reliability for Engineers and Scientists, Bilal M. Ayyub and Richard H. McCuen, 3rd edition, CRC press (2011).

Mode of Evaluation

Digital Assignments, Continuous Assessment Tests, Quiz, Final Assessment Test.

List of Experiments (Indicative)		
•	Introduction: Understanding Data types; importing/exporting data.	3 hours
•	Computing Summary Statistics /plotting and visualizing data using Tabulation and Graphical Representations.	3 hours
•	Applying correlation and simple linear regression model to real dataset; computing and interpreting the coefficient of determination.	3hours

•	Applying multiple linear regression model to real dataset; computing and interpreting the multiple coefficient of determination.	3 hours
•	Fitting the following probability distributions: Binomial distribution	3 hours
•	Normal distribution, Poisson distribution	3 hours
•	Testing of hypothesis for One sample mean and proportion from real-time problems.	3 hours
•	Testing of hypothesis for Two sample means and proportion from real-time problems	3 hours
•	Applying the t test for independent and dependent samples	2 hours
•	Applying Chi-square test for goodness of fit test and Contingency test to real dataset	2 hours
•	Performing ANOVA for real dataset for Completely randomized design, Randomized Block design ,Latin square Design	2 hours
	Total laboratory hours	**30 hours**

Mode of Evaluation			
Weekly Assessment, Final Assessment Test			
Recommended by Board of Studies	25-02-2017		
Approved by Academic Council	47	Date:	05-10-2017

MGT1022	LEAN START-UP MANAGEMENT	L	T	P	J	C
		1	0	0	4	2
Pre-requisite	**Nil**	**Syllabus version**				
		1.0				

Course Objectives:

To develop the ability to

1. Learn methods of company formation and management.
2. Gain practical skills in and experience of stating of business using pre-set collection of business ideas.
3. Learn basics of entrepreneurial skills.

Expected Course Outcome:

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

1. Understand developing business models and growth drivers
2. Use the business model canvas to map out key components of enterprise
3. Analyze market size, cost structure, revenue streams, and value chain
4. Understand build-measure-learn principles
5. Foreseeing and quantifying business and financial risks

Module: 1		**2hours**
Creativity and Design Thinking (identify the vertical for business opportunity, understand your customers, accurately assess market opportunity)		
Module: 2		**3 hours**
Minimum Viable Product (Value Proposition, Customer Segments, Build-measure-learn process)		
Module: 3		**3hours**
Business Model Development (Channels and Partners, Revenue Model and streams, Key Resources, Activities and Costs, Customer Relationships and Customer Development Processes, Business model canvas–the lean model-templates)		
Module: 4		**3 hours**
Business Plan and Access to Funding (visioning your venture, taking the product / service to market, Market plan including Digital & Viral Marketing, start-up finance – Costs / Profits & Losses / cash flow, Angel / VC / Bank Loans and Key elements of raising money)		
Module: 5		**2hours**
Legal, Regulatory, CSR, Standards,Taxes		
Module: 6	**Contemporary issues**	**2 hours**
Lectures by Entrepreneurs		
Total Lecture hours		**15 hours**

Text Book (s)

1.	Steve Blank, K & S Ranch (2012)The Startup Owner's Manual: The Step-By-Step Guide for Building a Great Company, 1st edition
2.	Steve Blank (2013) The Four Steps to the Epiphany, K&S Ranch; 2nd edition
3.	Eric Ries (2011) The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses, Crown Business

Reference Books	
1.	Holding a Cat by the Tail, Steve Blank, K & S Ranch Publishing LLC (August 14, 2014)
2.	Product Design and Development, Karal TUlrich, SDEppinger, McGrawHill
3.	Zero to One: Notes on Startups, or How to Build the Future, Peter Thiel, Crown Business (2014)
4.	Lean Analytics: Use Data to Build a Better Startup Faster (Lean Series), Alistair Croll & Benjamin Yoskovitz, O' Reilly Media; 1st Edition (March 21, 2013)
5.	Inspired: How to create Products Customers Love, Marty Cagan,S VPG Press; 1st edition (June18, 2008)
6.	**Website References:** 1. http://theleanstartup.com/ 2. https://www.kickstarter.com/projects/881308232/only-on-kickstarter-the-leaders-guide-by-eric-ries 3. http://businessmodelgeneration.com/ 4. https://www.leanstartupmachine.com/ 5. https://www.youtube.com/watch?v=fEvKo90qBns 6. http://thenextweb.com/entrepreneur/2015/07/05/whats-wrong-with-the-lean-startup-methodology/#gref 7. http://www.businessinsider.in/Whats-Lean-about-Lean-Startup/articleshow/53615661.cms 8. https://steveblank.com/tools-and-blogs-for-entrepreneurs/ 9. https://hbr.org/2013/05/why-the-lean-start-up-changes-everything 10. chventures.blogspot.in/platformsandnetworks.blogspot.in/p/saas-model.html
Teaching Modes: Assignments; Field Trips, Case Studies; e-learning; Learning through research, TED Talks	

Project				
1.	Project	60 hours		
Total Project		**60 hours**		
Recommended by Board of Studies		08.06.2015		
Approved by Academic Council		37th ACM	**Date**	16.06.2015

PHY1701	ENGINEERING PHYSICS	L	T	P	J	C
		3	0	2	0	4
Pre-requisite	**Physics of 12th standard or equivalent**	**Syllabus version**				
		1.0				

Course Objectives:

To enable the students to understand the basics of the latest advancements in Physics viz., Quantum Mechanics, Nanotechnology, Lasers, Electro Magnetic Theory and Fiber Optics.

Expected Course Outcome:

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

1. To understand the dual nature of radiation and matter.
2. To apply Schrodinger‘s equations to solve finite and infinite potential problems.
3. To apply quantum ideas at the nanoscale.
4. To apply quantum ideas for understanding the operation and working principle of optoelectronic devices.
5. To analyze the Maxwell‘s equations in differential and integral form.
6. To classify the optical fiber for different Engineering applications.
7. To apply concept of Lorentz Transformation for engineering applications.
8. To demonstrate the quantum mechanical ideas – Lab

Module: 1	**Introduction to Modern Physics**	**6 hours**

Planck‘s concept (hypothesis), Compton Effect, Particle properties of wave: Matter Waves, Davisson Germer Experiment, Heisenberg Uncertainty Principle, Wave function, and Schrodinger equation (time dependent & independent).

Module: 2	**Applications of Quantum Physics**	**5 hours**

Particle in a 1-D box (Eigen Value and Eigen Function), 3-D Analysis (Qualitative), Tunneling Effect (Qualitative) (AB 205), Scanning Tunneling Microscope (STM).

Module: 3	**Nanophysics**	**5 hours**

Introduction to Nano-materials, Moore‘s law, Properties of Nano-materials, Quantum confinement, Quantum well, wire & dot, Carbon Nano-tubes (CNT), Applications of nanotechnology in industry.

Module: 4	**Laser Principles and Engineering Application**	**6 hours**

Laser Characteristics, Spatial and Temporal Coherence, Einstein Coefficient & its significance, Population inversion, Two, three & four level systems, Pumping schemes, Threshold gain coefficient, Components of laser, Nd-YAG, He-Ne, CO2 and Dye laser and their engineering applications.

Module: 5	**Electromagnetic Theory and its application**	**6 hours**

Physics of Divergence, Gradient and Curl, Qualitative understanding of surface and volume integral, Maxwell Equations (Qualitative), Wave Equation (Derivation), EM Waves, Phase velocity, Group velocity, Group index , Wave guide (Qualitative)

Module: 6	**Propagation of EM waves in Optical fibers and Optoelectronic Devices**	**6 hours**

Light propagation through fibers, Acceptance angle, Numerical Aperture, Types of fibers - step index, graded index, single mode & multimode, Attenuation, Dispersion-intermodal and intramodal. Sources-LED & Laser Diode, Detectors-Photodetectors- PN & PIN - Applications of fiber optics in communication- Endoscopy.

Module: 7	Special Theory of Relativity	**9 hours**

Frame of reference, Galilean relativity, Postulate of special theory of relativity, Simultaneity, length contraction and time dilation.

Module: 8	**Contemporary issues**	**2 hours**
Lecture by Industry Experts		
Total Lecture hours		**45 hours**
Text Book (s)		
1.	Arthur Beiser et al., Concepts of Modern Physics, 2013, Sixth Edition, Tata McGraw Hill. William Silfvast,	
2.	Laser Fundamentals, 2008, Cambridge University Press	
3.	D. J. Griffith, Introduction to Electrodynamics, 2014, 4th Edition, Pearson	
4.	Djafar K. Mynbaev and Lowell L.Scheiner, Fiber Optic Communication Technology, 2011, Pearson	
Reference Books		
1.	Raymond A. Serway, Clement J. Mosses, Curt A. Moyer Modern Physics, 2010, 3rd Indian Edition Cengage learning.	
2.	John R. Taylor, Chris D. Zafiratos and Michael A. Dubson, Modern Physics for Scientists and Engineers, 2011, PHI Learning Private Ltd.	
3.	Kenneth Krane Modern Physics, 2010, Wiley Indian Edition.	
4.	Nityanand Choudhary and RichaVerma, Laser Systems and Applications, 2011, PHI Learning Private Ltd.	
5.	S. Nagabhushana and B. Sathyanarayana, Lasers and Optical Instrumentation, 2010, I.K. International Publishing House Pvt. Ltd.	
6.	R. Shevgaonkar, Electromagnetic Waves, 2005, 1st Edition, Tata McGraw Hill	
7.	Principles of Electromagnetics, Matthew N.O. Sadiku, 2010, Fourth Edition, Oxford	
8.	Ajoy Ghatak and K. Thyagarajan, Introduction to Fiber Optics, 2010, Cambridge University Press	
Mode of Evaluation: Quizzes , Digital Assignments, CAT-I and II and FAT		
List of Challenging Experiments (Indicative)		
1.	Determination of Planck‘s constant using electrolumine scence process	2 hrs
2.	Electron diffraction	2 hrs
3.	Determination of wave length of laser source (He-Ne laser and diodelasers of Different wave lengths) using diffraction technique	2 hrs
4.	Determination of size offine particle using laser diffraction	2 hrs
5.	Determination of the track width (periodicity) in a written CD	2 hrs
6.	Optical Fiber communication (source+optical fiber+detector)	2 hrs
7.	Analysis of crystallite size and strain in a nano-crystalline film using X-ray diffraction	2 hrs
8.	Numerical solutions of Schrödinger equation (e.g. particle in a box problem) (can be given as an assignment)	2 hrs
9.	Laser coherence length measurement	2 hrs

10.	Proof for transverse nature of E.M. waves	2 hrs
11.	Quantum confinement and Heisenberg's uncertainty principle	2 hrs
12.	Determination of angle of prism and refractive index for various colour – Spectrometer	2 hrs
13.	Determination of divergence of a laser beam	2 hrs
14.	Determination of crystalline size for nanomaterial (Computer simulation)	2 hrs
15.	Demonstration of phase velocity and group velocity (Computer simulation)	2 hrs
Total Laboratory Hours		**30 hours**

Mode of assessment: CAT / FAT			
Recommended by Board of Studies	04.06.2019		
Approved by Academic Council	46th ACM	**Date**	24.08.2017

PHY1901	INTRODUCTION TO INNOVATIVE PROJECTS	L	T	P	J	C
		1	0	0	0	1
Pre-requisite	Nil	Syllabus version				
		1.0				

Course Objectives:

This course is offered to the students in the 1st Year of B. Tech. in order to orient them towards independent, systemic thinking and be innovative.

1. To make students confident enough to handle the day to day issues.
2. To develop the ‒Thinking Skill‖ of the students, especially Creative Thinking Skills
3. To train the students to be innovative in all their activities
4. To prepare a project report on a socially relevant theme as a solution to the existing issues

Expected Course Outcome:

1. To understand the various types of thinking skills.
2. To enhance the innovative and creative ideas.
3. To find out a suitable solution for socially relevant issues-J component

Module: 1A	Self Confidence	1 hour

Understanding self – Johari Window – SWOT Analysis – Self Esteem – Being a contributor – CaseStudy

Project : Exploring self, understanding surrounding, thinking about how s(he) can be acontributor Forthe society, Creating a big picture of being an innovator–writing a 1000 words imaginary Autobiography of self–Topic ‒Mr. X–the great innovator of 2015‖ and upload. (**non-contact hours)**

Module: 1B	Thinking Skill	1 hour

Thinking and Behaviour–Types of thinking–Concrete– Abstract, Convergent, Divergent, Creative, Analytical, Sequential and Holistic thinking–Chunking Triangle–Context Grid – Examples – CaseStudy.

Project: Meeting atleast 50 people belonging to various strata of life and talk to them / make field visits to identify a min. of 100 society related issues, problems for which they need solutions and categories them and upload along with details of people met and lessons learnt. **(4 non-contact hours)**

Module: 1C	Lateral ThinkingSkill	1 hour

Blooms Taxonomy–HOTS–Out of the box thinking–de Bono lateral thinking model–Examples

Project : Last weeks-incomplete portion to be done and uploaded

Module: 2A	Creativity	1 hour

Creativity Models–Walla–Barrons–Koberg & Begnall–Examples

Project: Selecting 5 out of 100 issues identified for future work. Criteria based approach for prioritisation, use of statistical tools & upload. **(4 non-contact hours)**

Module: 2B	Brain storming	1 hour
25 brainstorming techniques and examples **Project:** Brainstorm and come out with as many solutions as possible for the top 5 issues identified & upload. **(4 non-contact hours)**		
Module: 3	**Mind Mapping**	**1 hour**
Mind Mapping techniques and guidelines. Drawing a mind map **Project:** Using Mind Maps get another set of solutions for the next 5 issues (issue 6–10). **(4 non-contact hours)**		
Module: 4A	**Systems thinking**	**1 hour**
Systems Thinking essentials–examples–Counter Intuitive condemns **Project:** Select 1 issue / problem for which the possible solutions are available with you. Apply Systems Thinking process and pick up one solution [explanation should be given why the other possible solutions have been left out].Goback to the customer and assess the acceptability and upload. **(4 non-contact hours)**		
Module: 4B	**Design Thinking**	**1 hour**
Design thinking process–Human element of design thinking– case study **Project:** Apply design thinking to the selected solution; apply the engineering & scientific tinge to it. Participate in ―design week‖ celebration sup load the weeks learning out come.		
Module: 5A	**Innovation**	**1 hour**
Difference between Creativity and Innovation–Examples of innovation–Being innovative. **Project:** A literature searches on proto typing of your solution finalized. Prepare a proto type model or processand upload. **(4 non-contact hours)**		
Module: 5B	**Blocks for Innovation**	**1 hour**
Identify Blocks for creativity and innovation – overcoming obstacles – Case Study **Project:** Project presentation on problem identification, solution, innovations-expected results–Interim review with PPT presentation. **(4 non-contact hours)**		
Module: 5C	**Innovation Process**	**1 hour**
Steps for Innovation–right climate for innovation **Project:** Refining the project, based on the review report and uploading the text. **(4 non-contact hours)**		
Module: 6A	**Innovation in India**	**1 hour**
Stories of 10 Indian innovations **Project:** Making the project better with add ons. **(4 non- contact hours)**		
Module: 6B	**JUGAAD Innovation**	**1 hour**
Frugal and flexible approach to innovation-doing more with less Indian Examples **Project:** Fine tuning the innovation project with JUGAAD principles and uploading (Credit for JUGAAD implementation). **(4 non-contact hours)**		
Module: 7A	**Innovation Project Proposal Presentation**	**1 hour**
Project proposal contents, economicinput, ROI–Template **Project:** Presentation of the innovative project proposal and upload. **(4 non- contact hours)**		
Module: 8A	**Contemporary issues**	**1 hour**
Contemporary issue in Innovation **Project:** Final project Presentation, Vivavoce Exam (**4 non-contact hours)**		

Total Lecture hours				**15 hours**
Text Book(s)				
1.	How to have Creative Ideas, Edward debone,Vermil on publication, UK, 2007			
2.	The Art of Innovation, Tom Kelley & Jonathan Littman, Profile Books Ltd., UK, 2008			
Reference Books				
1.	Creating Confidence, Meribeth Bonct, Kogan Page India Ltd., New Delhi, 2000			
2.	Lateral Thinking Skills, Paul Sloane, Keogan Page India Ltd, New Delhi, 2008			
3.	Indian Innovators, Akhat Agrawal, Jaico Books, Mumbai, 2015			
4.	JUGAAD Innovation, Navi Radjou, Jaideep Prabhu, Simone Ahuja Random house India, Noida, 2012.			
Mode of Evaluation: CAT / Assignment / Quiz / FAT / Project / Seminar Three reviews with weightage of 25 : 25 : 50 along with reports				
Recommended by Board of Studies	15.12.2015			
Approved by Academic Council	39th ACM	**Date**	17.12.2015	

Course code	Technical Answers for Real World Problems (TARP)	L	T	P	J	C
SWE1901		1	0	0	4	2
Pre-requisite	PHY1901	Syllabus version				
		v. 1.0				

Course Objectives:

- To help students to identify the need for developing newer technologies for industrial / societal needs
- To train students to propose and implement relevant technology for the development of the prototypes / products
- To make the students learn to the use the methodologies available to assess the developed prototypes / products

Expected Course Outcome:

At the end of the course, the student will be able to
[1] Identify real life problems related to society
[2] Apply appropriate technology(ies) to address the identified problems using engineering principles and arrive at innovative solutions

Module:1		15 hours

1. Identification of real life problems
2. Field visits can be arranged by the faculty concerned
3. 6 – 10 students can form a team (within the same / different discipline)
4. Minimum of eight hours on self-managed team activity
5. Appropriate scientific methodologies to be utilized to solve the identified issue
6. Solution should be in the form of fabrication/coding/modeling/product design/process design/relevant scientific methodology(ies)
7. Consolidated report to be submitted for assessment
8. Participation, involvement and contribution in group discussions during the contact hours will be used as the modalities for the continuous assessment of the theory component
9. Project outcome to be evaluated in terms of technical, economical, social, environmental, political and demographic feasibility
10. Contribution of each group member to be assessed
11. The project component to have three reviews with the weightage of 20:30:50

Mode of Evaluation: (No FAT) Continuous Assessment the project done – Mark weightage of 20:30:50 – project report to be submitted, presentation and project reviews

Recommended by Board of Studies	28-02-2016		
Approved by Academic Council	No. 37	Date	16-06-2015

SWE1902	Industrial Internship	L	T	P	J	C
		0	0	0	0	1
Pre-requisite	Completion of minimum of Two semesters					

Course Objectives:

1. The course is designed so as to expose the students to industry environment and to take up on-site assignment as trainees or interns.

Expected Course Outcome:

1. Have an exposure to industrial practices and to work in teams
2. Communicate effectively
3. Understand the impact of engineering solutions in a global, economic, environmental and societal context
4. Develop the ability to engage in research and to involve in life-long learning
5. Comprehend contemporary issues
6. Engage in establishing his/her digital footprint

Contents	**4**	**Weeks**

Four weeks of work at industry site.

Supervised by an expert at the industry.

Mode of Evaluation: Internship Report, Presentation and Project Review

Recommended by Board of Studies	28-02-2016		
Approved by Academic Council	No. 37th	Date	16-06-2015

SWE1903	Comprehensive Examination	L	T	P	J	C
		0	0	0	0	2
Pre-requisite		Syllabus version				
		1.00				

Digital Logic and Microprocessor

Simplification of Boolean functions using K-Map – Combinational logic: Adder, subtractor, encoder, decoder, multiplexer, de-multiplexer – Sequential Logic: Flip flops- 8086 Microprocessor: instructions – peripherals: 8255, 8254, 8257.

Computer Architecture and Organization

Instructions - Instruction types- Instruction Formats - Addressing Modes- Pipelining- Data Representation - Memory Hierarchy- Cache memory-Virtual Memory- I/O Fundamentals- I/O Techniques - Direct Memory Access - Interrupts-RAID architecture

Programming, Data Structures and Algorithms

Programming in C; Algorithm Analysis – Iterative and Recursive Algorithms; ADT - Stack and its Applications - Queue and its Applications; Data Structures – Arrays and Linked Lists; Algorithms - Sorting – Searching; Trees – BST, AVL; Graphs – BFS , DFS , Dijkstra‘s Shortest Path Algorithm.

Theory of Computation

Deterministic Finite Automata, Non deterministic Finite Automata, Regular Expressions, Context Free Grammar, Push down Automata and Context Free Languages, Turing Machines.

Web Technologies

Web Architecture- JavaScript – objects String, date, Array, Regular Expressions, DHTML-HTML DOM Events; Web Server – HTTP- Request/Response model-RESTful methods- State Management – Cookies , Sessions – AJAX.

Operating Systems

Processes, Threads, Inter-process communication, CPU scheduling, Concurrency and synchronization, Deadlocks, Memory management and Virtual memory & File systems.

Database Management System

DBMS, Schema, catalog, metadata, data independence, pre-compiler; Users-naïve, sophisticated, casual ;ER Model- Entity, attributes, structural constraints; Relational Model-Constraints, Relational Algebra operations; SQL- DDL, DML, TCL, DCL commands, basic queries and Top N queries; Normalization-properties, 1NF, 2NF, 3NF, BCNF; Indexing-different types, Hash Vs B-tree Index; Transaction-problems, Concurrency Control-techniques, Recovery-methods.

Data Communication and Computer Networks

Circuit Switching, Packet Switching, Frame Relay, Cell Switching, ATM , OSI Reference model, TCP\IP, Network topologies, LAN Technologies, Error detection and correction techniques, Internet protocols , IPv4/IPv6, Routing algorithms, TCP and UDP, Sockets, Congestion control, Application Layer Protocols, Network Security: Basics of public and private key cryptosystems-Digital Signatures and Hash codes, Transport layer security, VPN, Firewalls.

Recommended by Board of Studies	05-03-2016

SWE1904	Masters Thesis	L	T	P	J	C
		0	0	0	0	16
Pre-requisite	As per the academic regulations	Syllabus version				
		1.0				

Course Objectives:

To provide sufficient hands-on learning experience related to the design, development and analysis of suitable product / process so as to enhance the technical skill sets in the chosen field and also to give research orientation

Expected Course Outcome:

At the end of the course the student will be able to

1. Formulate specific problem statements for ill-defined real life problems with reasonable assumptions and constraints.
2. Perform literature search and / or patent search in the area of interest.
3. Conduct experiments / Design and Analysis / solution iterations and document the results.
4. Perform error analysis / benchmarking / costing
5. Synthesise the results and arrive at scientific conclusions / products / solution
6. Document the results in the form of technical report / presentation

Contents

1. Capstone Project may be a theoretical analysis, modeling & simulation, experimentation & analysis, prototype design, fabrication of new equipment, correlation and analysis of data, software development, applied research and any other related activities.
2. Project can be for two semesters based on the completion of required number of credits as per the academic regulations.
3. Should be individual work.
4. Carried out inside or outside the university, in any relevant industry or research institution.
5. Publications in the peer reviewed journals / International Conferences will be an added advantage

Mode of Evaluation: Periodic reviews, Presentation, Final oral viva, Poster submission

Recommended by Board of Studies	10.06.2016		
Approved by Academic Council	41st AC	Date	17.06.2016

CHY1002	Environmental Sciences	L	T	P	J	C
		3	0	0	0	3

Pre-requisite		Syllabus version
		1.1

Course Objectives:

1. To make students understand and appreciate the unity of life in all its forms, the implications of life style on the environment.
2. To understand the various causes for environmental degradation.
3. To understand individuals contribution in the environmental pollution.
4. To understand the impact of pollution at the global level and also in the local environment.

Expected Course Outcome:

1. Students will recognize the environmental issues in a problem oriented interdisciplinary perspectives
2. Students will understand the key environmental issues, the science behind those problems and potential solutions.
3. Students will demonstrate the significance of biodiversity and its preservation
4. Students will identify various environmental hazards
5. Students will design various methods for the conservation of resources
6. Students will formulate action plans for sustainable alternatives that incorporate science, humanity, and social aspects
7. Students will have foundational knowledge enabling them to make sound life decisions as well as enter a career in an environmental profession or higher education.

Module:1	Environment and Ecosystem	7 hours

Key environmental problems, their basic causes and sustainable solutions. IPAT equation. Ecosystem, earth – life support system and ecosystem components; Food chain, food web, Energy flow in ecosystem; Ecological succession- stages involved, Primary and secondary succession, Hydrarch, mesarch, xerarch; Nutrient, water, carbon, nitrogen, cycles; Effect of human activities on these cycles.

Module:2	Biodiversity	6 hours

Importance, types, mega-biodiversity; Species interaction - Extinct, endemic, endangered and rare species; Hot-spots; GM crops- Advantages and disadvantages; Terrestrial biodiversity and Aquatic biodiversity – Significance, Threats due to natural and anthropogenic activities and Conservation methods.

Module:3	Sustaining Natural Resources and Environmental Quality	7 hours

Environmental hazards – causes and solutions. Biological hazards – AIDS, Malaria, Chemical hazards- BPA, PCB, Phthalates, Mercury, Nuclear hazards- Risk and evaluation of hazards. Water footprint; virtual water, blue revolution. Water quality management and its conservation. Solid and hazardous waste – types and waste management methods.

Module:4	**Energy Resources**	**6 hours**
Renewable - Non renewable energy resources- Advantages and disadvantages - oil, Natural gas, Coal, Nuclear energy. Energy efficiency and renewable energy. Solar energy, Hydroelectric power, Ocean thermal energy, Wind and geothermal energy. Energy from biomass, solar- Hydrogen revolution.		
Module:5	**Environmental Impact Assessment**	**6 hours**
Introduction to environmental impact analysis. EIA guidelines, Notification of Government of India (Environmental Protection Act – Air, water, forest and wild life). Impact assessment methodologies. Public awareness. Environmental priorities in India.		
Module:6	**Human Population Change and Environment**	**6 hours**
Urban environmental problems; Consumerism and waste products; Promotion of economic development – Impact of population age structure – Women and child welfare, Women empowerment. Sustaining human societies: Economics, environment, policies and education.		
Module:7	**Global Climatic Change and Mitigation**	**5 hours**
Climate disruption, Green house effect, Ozone layer depletion and Acid rain. Kyoto protocol, Carbon credits, Carbon sequestration methods and Montreal Protocol. Role of Information technology in environment-Case Studies.		
Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Books	
1.	G. Tyler Miller and Scott E. Spoolman (2016), Environmental Science, 15th Edition, Cengage learning.
2.	George Tyler Miller, Jr. and Scott Spoolman (2012), Living in the Environment – Principles, Connections and Solutions, 17th Edition, Brooks/Cole, USA.
Reference Books	
1.	David M.Hassenzahl, Mary Catherine Hager, Linda R.Berg (2011), Visualizing Environmental Science, 4thEdition, John Wiley & Sons, USA.

Mode of evaluation: Internal Assessment (CAT, Quizzes, Digital Assignments) & FAT			
Recommended by Board of Studies	12.08.2017		
Approved by Academic Council	No. 46	Date	24.08.2017