Sri Sivasubramaniya Nadar College of Engineering, Kalavakkam - 603110

(An Autonomous Institution, Affiliated to Anna University, Chennai)

REGULATIONS 2018

CHOICE BASED CREDIT SYSTEM

B.E. COMPUTER SCIENCE AND ENGINEERING

CURRICULUM

SEMESTER I

S. No	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
THEORY								
1.	UEN1176	Communicative English	HS	3	3	0	0	3
2.	UMA1176	Algebra and Calculus	BS	5	3	2	0	4
3.	UPH1176	Engineering Physics	BS	3	3	0	0	3
4.	UCY1176	Engineering Chemistry	BS	3	3	0	0	3
5.	UGE1176	Problem Solving and Programming in Python	ES	3	3	0	0	3
6.	UGE1177	Engineering Graphics	ES	5	1	0	4	3
PRACTICALS								
7.	UGE1197	Programming in Python Lab	ES	3	0	0	3	1.5
8.	UGS1197	Physics and Chemistry Lab	BS	3	0	0	3	1.5
			TOTAL	**28**	**16**	**2**	**10**	**22**

SEMESTER II

S. No	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
THEORY								
1.	UEN1276	Technical English	HS	3	3	0	0	3
2.	UMA1276	Complex Functions and Laplace Transforms	BS	5	3	2	0	4
3.	UPH1276	Physics for Information Science	BS	3	3	0	0	3
4.	UCY1276	Environmental Science	HS	3	3	0	0	3
5.	UEE1276	Basic Electrical, Electronics and Measurement Engineering	ES	4	3	1	0	3.5
6.	UCS1201	Programming in C	PC	4	3	1	0	3.5
PRACTICALS								
7.	UGE1297	Design Thinking and Engineering Practices Lab	ES	3	0	0	3	1.5
8.	UCS1211	Programming in C Lab	PC	3	0	0	3	1.5
			TOTAL	**28**	**18**	**4**	**6**	**23**

SEMESTER III

S. No	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
THEORY								
1.	UMA1377	Discrete Mathematics	BS	5	3	2	0	4
2.	UCS1301	Digital Principles and System Design	PC	3	3	0	0	3
3.	UCS1302	Data Structures	PC	3	3	0	0	3
4.	UCS1303	Object Oriented Programming using Java	PC	3	3	0	0	3
5.	UCS1304	UNIX and Shell Programming	PC	4	2	0	2	3
6.	UEC1351	Principles of Communication Engineering	ES	3	3	0	0	3
PRACTICALS								
7.	UCS1311	Digital Design Lab	PC	3	0	0	3	1.5
8.	UCS1312	Data Structures Lab	PC	4	0	0	4	2
9.	UCS1313	Object Oriented Programming using Java Lab	PC	3	0	0	3	1.5
			TOTAL	**31**	**17**	**2**	**12**	**24**

SEMESTER IV

S. No	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
THEORY								
1.	UMA1478	Probability and Statistics	BS	5	3	2	0	4
2.	UCS1401	Computer Organization and Architecture	PC	3	3	0	0	3
3.	UCS1402	Operating Systems	PC	3	3	0	0	3
4.	UCS1403	Design and Analysis of Algorithms	PC	5	3	0	2	4
5.	UCS1404	Database Management Systems	PC	3	3	0	0	3
6.	UCS1405	Software Engineering	PC	3	3	0	0	3
PRACTICALS								
7.	UCS1411	Operating Systems Lab	PC	3	0	0	3	1.5
8.	UCS1412	Database Lab	PC	3	0	0	3	1.5
9.	UEN1498	Interpersonal Skills	EEC	2	0	0	2	1
			TOTAL	**30**	**18**	**2**	**10**	**24**

SEMESTER V

S. No	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
THEORY								
1.	UCS1501	Computer Networks	PC	3	3	0	0	3
2.	UCS1502	Microprocessors and Interfacing	PC	3	3	0	0	3
3.	UCS1503	Theory of Computation	PC	5	3	2	0	4
4.	UCS1504	Artificial Intelligence	PC	5	3	0	2	4
5.	UCS1505	Introduction to Cryptographic Techniques	PC	3	3	0	0	3
6.		Professional Elective – 1	PE	3	3	0	0	3
PRACTICALS								
7.	UCS1511	Networks Lab	PC	3	0	0	3	1.5
8.	UCS1512	Microprocessors Lab	PC	3	0	0	3	1.5
			TOTAL	**26**	**18**	**2**	**6**	**23**

SEMESTER VI

S. No	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
THEORY								
1.	UCS1601	Internet Programming	PC	3	3	0	0	3
2.	UCS1602	Compiler Design	PC	5	3	0	2	4
3.	UCS1603	Introduction to Machine Learning	PC	3	3	0	0	3
4.	UCS1604	Object Oriented Analysis and Design	PC	3	3	0	0	3
5.		Professional Elective – 2	PE	3	3	0	0	3
6.		Open Elective - 1	OE	3	3	0	0	3
PRACTICALS								
7.	UCS1611	Internet Programming Lab	PC	3	0	0	3	1.5
8.	UCS1617	Mini Project	EEC	3	0	0	3	1.5
			TOTAL	**26**	**18**	**0**	**8**	**22**

SEMESTER VII

S. No	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
THEORY								
1.	UCS1701	Distributed Systems	PC	3	3	0	0	3
2.	UCS1702	Mobile Computing	PC	3	3	0	0	3
3.	UCS1703	Graphics and Multimedia	PC	3	3	0	0	3
4.	UCS1704	Management and Ethical Practices	HS	3	3	0	0	3
5.		Professional Elective - 3	PE	3	3	0	0	3
6.		Professional Elective - 4	PE	3	3	0	0	3
PRACTICALS								
7.	UCS1711	Mobile Application Development Lab	PC	3	0	0	3	1.5
8.	UCS1712	Graphics and MultimediaLab	PC	3	0	0	3	1.5
			TOTAL	**24**	**18**	**0**	**6**	**21**

SEMESTER VIII

S. No	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
THEORY								
1.		Professional Elective - 5	PE	3	3	0	0	3
2.		Open Elective – 2	OE	3	3	0	0	3
PRACTICALS								
3.	UCS1818	Project Work	EEC	18	0	0	18	9
			TOTAL	**24**	**6**	**0**	**18**	**15**

TOTAL NO. OF CREDITS: 174

HUMANITIES AND SOCIAL SCIENCES (HS)

S. No	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
THEORY								
1.	UEN1176	Communicative English	HS	3	3	0	0	3
2.	UEN1276	Technical English	HS	3	3	0	0	3
3.	UCY1276	Environmental Science	HS	3	3	0	0	3
4.	UCS1704	Management and Ethical Practices	HS	3	3	0	0	3

BASIC SCIENCES (BS)

S. No	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
THEORY								
1.	UMA1176	Algebra and Calculus	BS	4	4	0	0	4
2.	UPH1176	Engineering Physics	BS	3	3	0	0	3
3.	UCY1176	Engineering Chemistry	BS	3	3	0	0	3
4.	UGS1197	Physics and Chemistry Lab	BS	3	0	0	3	1.5
5.	UMA1276	Complex Functions and Laplace Transforms	BS	5	3	2	0	4
6.	UPH1276	Physics for Information Science	BS	3	3	0	0	3
7.	UMA1377	Discrete Mathematics	BS	5	3	2	0	4
8.	UMA1478	Probability and Statistics	BS	5	3	2	0	4

ENGINEERING SCIENCES (ES)

S. No	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
THEORY								
1.	UGE1176	Problem Solving and Programming in Python	ES	3	3	0	0	3
2.	UGE1177	Engineering Graphics	ES	5	1	0	4	3
3.	UGE1197	Programming in Python Lab	ES	3	0	0	3	1.5
4.	UEE1276	Basic Electrical, Electronics and Measurements Engineering	ES	4	3	1	0	3.5
5.	UGE1297	Design Thinking and Engineering Practices Lab	ES	3	0	0	3	1.5
6.	UEC1351	Principles of Communication Engineering	ES	3	3	0	0	3

PROFESSIONAL CORE (PC)

S. No	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
THEORY								
1.	UCS1201	Programming in C	PC	4	3	1	0	3.5
2.	UCS1211	Programming in C Lab	PC	3	3	0	0	1.5
3.	UCS1301	Digital Principles and System Design	PC	3	3	0	0	3
4.	UCS1302	Data Structures	PC	3	3	0	0	3
5.	UCS1303	Object Oriented Programming using Java	PC	3	3	0	0	3
6.	UCS1304	UNIX and Shell Programming	PC	4	2	0	2	3
7.	UCS1311	Digital Design Lab	PC	3	0	0	3	1.5
8.	UCS1312	Data Structures Lab	PC	4	0	0	4	2
9.	UCS1313	Object Oriented Programming using Java Lab	PC	3	0	0	3	1.5
10.	UCS1401	Computer Organization and Architecture	PC	3	3	0	0	3
11.	UCS1402	Operating Systems	PC	3	3	0	0	3
12.	UCS1403	Design and Analysis of Algorithms	PC	5	3	0	2	4
13.	UCS1404	Database Management Systems	PC	3	3	0	0	3
14.	UCS1405	Software Engineering	PC	3	3	0	0	3
15.	UCS1411	Operating Systems Lab	PC	3	0	0	3	1.5
16.	UCS1412	Database Lab	PC	3	0	0	3	1.5
17.	UCS1501	Computer Networks	PC	3	3	0	0	3
18.	UCS1502	Microprocessors and Interfacing	PC	3	3	0	0	3
19.	UCS1503	Theory of Computing	PC	5	3	2	0	4
20.	UCS1504	Artificial Intelligence	PC	5	3	0	2	4
21.	UCS1505	Introduction to Cryptographic Techniques	PC	3	3	0	0	3
22.	UCS1511	Networks Lab	PC	3	0	0	3	1.5
23.	UCS1512	Microprocessors Lab	PC	3	0	0	3	1.5
24.	UCS1601	Internet Programming	PC	3	3	0	0	3
25.	UCS1602	Compiler Design	PC	5	3	0	2	4
26.	UCS1603	Introduction to Machine Learning	PC	3	3	0	0	3
27.	UCS1604	Object Oriented Analysis and Design	PC	3	3	0	0	3
28.	UCS1611	Internet Programming Lab	PC	3	0	0	3	1.5
29.	UCS1701	Distributed Systems	PC	3	3	0	0	3

30.	UCS1702	Mobile Computing	PC	3	3	0	0	3
31.	UCS1703	Graphics and Multimedia	PC	3	3	0	0	3
32.	UCS1711	Mobile Application Development Lab	PC	3	0	0	3	1.5
33.	UCS1712	Graphics and Multimedia Lab	PC	3	0	0	3	1.5

PROFESSIONAL ELECTIVES (PE)

SEMESTER V

ELECTIVE – I

S. No	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
THEORY								
1.	UCS1521	Advanced Database Management Systems	PE	3	3	0	0	3
2.	UCS1522	Software Testing	PE	3	3	0	0	3
3.	UCS1523	Digital Signal Processing	PE	3	3	0	0	3
4.	UCS1524	Logic Programming	PE	3	3	0	0	3
5.	UCS1525	Wireless and Adhoc Networks	PE	3	3	0	0	3
6.	UCS1526	Programming Paradigms	PE	3	3	0	0	3

SEMESTER VI

ELECTIVE – II

S. No	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
THEORY								
1.	UCS1621	Advanced Java Programming	PE	3	3	0	0	3
2.	UCS1622	Software Project Management	PE	3	3	0	0	3
3.	UCS1623	Image Processing and Analysis	PE	3	3	0	0	3
4.	UCS1624	Internet of Things	PE	3	3	0	0	3
5.	UCS1625	Foundations of Data Science	PE	3	3	0	0	3
6.	UCS1626	Cloud Computing	PE	3	3	0	0	3

SEMESTER VII

ELECTIVE – III

S. No	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
THEORY								
1.	UCS1721	Service Oriented Architecture	PE	3	3	0	0	3
2.	UCS1722	Social Network Analysis	PE	3	3	0	0	3
3.	UCS1723	Deep Learning	PE	3	3	0	0	3
4.	UCS1724	Multicore Architecture and Programming	PE	3	3	0	0	3
5.	UCS1725	Logic in Computer Science	PE	3	3	0	0	3
6.	UCS1726	Software Architecture	PE	3	3	0	0	3

ELECTIVE - IV

S. No	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
THEORY								
1.	UCS1727	GPU Computing	PE	3	3	0	0	3
2.	UCS1728	User Experience Design	PE	3	3	0	0	3
3.	UCS1729	Data Warehousing and Data Mining	PE	3	3	0	0	3
4.	UCS1731	Soft Computing	PE	3	3	0	0	3
5.	UCS1732	Embedded Systems	PE	3	3	0	0	3
6.	UCS1733	Graph Theory and Applications	PE	3	3	0	0	3

SEMESTER VIII

ELECTIVE - V

S. No	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
THEORY								
1.	UCS1821	Big Data Analytics	PE	3	3	0	0	3
2.	UCS1822	Agile Methodologies	PE	3	3	0	0	3
3.	UCS1823	Natural Language Processing	PE	3	3	0	0	3
4.	UCS1824	Robotics	PE	3	3	0	0	3
5.	UCS1825	Network and Server Security	PE	3	3	0	0	3
6.	UCS1826	Formal System Verification	PE	3	3	0	0	3

OPEN ELECTIVES (OFFERED TO OTHER DEPARTMENTS)

ODD SEMESTER

S. No	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
THEORY								
1.	UCS1941	Machine Learning Applications	OE	4	2	0	2	3
2.	UCS1942	Web Technology	OE	4	2	0	2	3

EVEN SEMESTER

S. No	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
THEORY								
1.	UCS1041	Introduction to Data Structures	OE	4	2	0	2	3
2.	UCS1042	Object Oriented Programming Techniques	OE	4	2	0	2	3
3.	UCS1043	Problem Solving and Programming in C	OE	4	2	0	2	3
4.	UCS1044	Introduction to Big Data Analytics	OE	4	2	0	2	3

EMPLOYABILITY ENHANCEMENT COURSES (EEC)

S. No	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
THEORY								
1.	UEN1498	Interpersonal Skills	EEC	2	0	0	2	1
2.	UCS1617	Mini Project	EEC	3	0	0	3	1.5
3.	UCS1818	Project Work	EEC	18	0	0	18	9

SUMMARY

S. No	SUBJECT AREA	CREDITS AS PER SEMESTER								CREDITS TOTAL	Percentage
		I	II	III	IV	V	VI	VII	VIII		
1.	HS	3	6	-	-	-	-	3	-	12	6.88
2.	BS	11.5	7	4	4	-	-	-	-	26.5	15.21
3.	ES	7.5	5	3	-	-	-	-	-	15.5	8.89
4.	PC	-	5	17	19	20	14.5	12	-	87.5	50.23
5.	PE	-	-	-	-	3	3	6	3	15	8.61
6.	OE	-	-	-	-	-	3	-	3	6	3.44
7.	EEC	-	-	-	1	-	1.5	-	9	11.5	6.60
	Total	22	23	24	24	23	22	21	15	174	
8.	Non-Credit / Mandatory										

COURSE CODE	COURSE TITLE	L	T	P	C
UEN1176	COMMUNICATIVE ENGLISH	3	0	0	3

OBJECTIVES

- To develop the basic reading and writing skills
- To help learners develop their listening skills, which will, enable them listen to lectures and comprehend them by asking questions and seeking clarifications.
- To help learners develop their speaking skills to enable them to speak fluently in real contexts.

UNIT I LANGUAGE FOR 'SMALL TALK' 9

Reading: Short comprehension passages, Practice in skimming-scanning and predicting. Writing: Completing sentences, Developing hints.Language Development: asking and answering - Wh- Questions and Yes / No questions. Vocabulary Development: Prefixes and Suffixes, Polite Expressions.Speaking: Introducing oneself, Exchanging personal information. Listening: Listening comprehension of short texts.

UNIT II DEVELOP WRITING THROUGH READING 9

Reading: Short narratives and descriptions from newspapers (including dialogues and conversations); Reading Comprehension Texts with varied question types.Writing: Paragraph writing (topic sentence, cohesive devices, organizational pattern). Language Development: prepositions, and clauses.Vocabulary Development: guessing meanings of words in context. Speaking: Informal conversations, chunking at right places.Listening: Listening to telephone conversations, short presentations and TV News.

UNIT III PREPARING FOR A CAREER 9

Reading: Short texts and longer passages (close reading). Writing: Reordering jumbled sentences. Language Development: Degrees of comparisons, pronouns. Vocabulary Development: Idioms and phrases. Speaking: Short presentations using power point slides. Listening: Listening to ted talks and long speeches for comprehension.

UNIT IV IMPROVING SPEAKING 9

Reading:Readingdifferenttypesoftexts(literary,journalistic,printmedia)for comprehension and pleasure. Writing: letter writing (informal or personal letters) and e-mails etiquette. Language Development: Tenses: simple present and past,present and past continuous Vocabulary Development: single word substitutes, collocations. Speaking: Role plays (literary and nonliterary texts). Listening: Listening comprehension (IELTS, TOEFL and others).

UNIT V LISTENING FOR DEEPER UNDERSTANDING 9

Reading: Reading for comparisons and contrast and other deeper levels of meaning. Writing: Writing short pieces – developing an outline, identifying main and subordinate ideas. Language Development: modal verbs, perfect tenses. Vocabulary Development: phrasal verbs, fixed and

semi-fixed expressions (including idioms), fillers Speaking: Group Discussions Listening: Listening to lectures and making notes.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Apply reading strategies to comprehend articles of a general kind in magazines and newspapers.
- Participate effectively in informal conversations; introduce themselves and their friends and express opinions in English.
- Comprehend conversations and short talksdelivered inEnglish inboth formaland informal contexts
- Write short essays of a general kind and personal letters and emails in English.

TEXTBOOKS

1. Board of Editors, Using English: A Course Book for Undergraduate Engineers and Technologists. Orient BlackSwan Limited, Hyderabad, 2015.

REFERENCE BOOKS

1. Richards, C. Jack.Interchange Students' Book-2 New Delhi, Cambridge University Press, 2015.
2. Bailey, Stephen. Academic Writing: A practical guide for students, New York: Rutledge, 2011.
3. Means,L.Thomas,andElaineLanglois,English&CommunicationforColleges. Cengage Learning, USA, 2007.
4. Redston, Chris & Gillies Cunningham Face2Face (Pre-intermediate Student 's Book& Workbook) Cambridge University Press, New Delhi, 2005.
5. Comfort, Jeremy, et al. Speaking Effectively: Developing Speaking Skills for Business English. Cambridge University Press, Cambridge: Reprint 2011.

COURSE CODE	COURSE TITLE	L	T	P	C
UMA1176	ALGEBRA AND CALCULUS	3	2	0	4

OBJECTIVES

The objective of this course is to enable the student to

- Understand De Moivre's Theorem and use it in finding the expansion of trigonometric functions.
- Evaluate the Eigen values and Eigen vectors and diagonalize the given matrix. Understand the concept of circle of curvature, evolute and envelope of a given curve. Familiarize the functions of two variables and finding its extreme points.
- Understand Beta and Gamma functions and their relations, evaluation of double integrals and triple integrals.

UNIT ITRIGNOMETRIC SERIES 12

De Moivre's Theorem (with proof) – Roots of a complex number, expansion of sin $n\theta$, cos $n\theta$andtan $n\theta$ in powers ofsin θ ,cosθ andtan θ . Addition formulae for any number of angles, Expansion of$sin^m\theta, cos^n\theta and sin^m\theta, cos^n\theta$in a series of sinesor cosines of multiples ofθ, Complex function – Exponential function of a complex variable, Hyperbolic functions, Real and imaginary parts of circular functions, Logarithmic function of complex variable.

UNIT IIMATRICES 12

Eigen values and Eigen vectors – Properties of Eigen values - Linear dependence and independence of eigen vectors - Cayley-Hamilton theorem (excluding proof), Reduction to Diagonal form – Similarity transformation, Quadratic form – Reduction of Quadratic form to canonical form, Nature of a Quadratic form, Complex Matrices.

UNIT III DIFFERENTIAL CALCULUS 12

Curvature – Cartesian and parametric coordinates, radius of curvature – Cartesian form (with proof) parametric and polar form, Centre of curvature and circle of curvature in Cartesian form, Evolute and envelope.

UNIT IV FUNCTIONS OF SEVERAL VARIABLES 12

Partial derivatives – Euler's theorem for homogenous functions – Total derivatives – Differentiation of implicit functions – Jacobians - Taylor's expansion – Maxima and Minima – Lagrangian method of undetermined multipliers, Differentiation under the integral sign.

UNIT V INTEGRAL CALCULUS 12

Beta andGammafunctions–Properties,TransformationofBetaandGammafunctions, Relation between Beta and Gamma functions, Double integrals, Change the order of Integration, Evaluation of double integrals in polar co-ordinations, Triple integrals.

TOTAL PERIODS: 60

OUTCOMES

On successful completion of this course, the student will be able to

- Obtain the expansion of trigonometric functions using De-Moivre's theorem.
- Determine the Eigen values and Eigen vectors and diagonalize the given matrix.
- Evaluate the circle of curvature, evolute and envelope of a given curve.
- Find Taylor's expansion for functions of two variables, solve problems using Jacobians
- and find the extreme points of a function of two variables.
- Solveproblemsusingbetaandgammafunctionsandevaluateproblemsindouble integral and triple integral.

TEXTBOOKS

1. Grewal B.S, Higher Engineering Mathematics, Khanna Publishers, 43rd Edition, 2016.
2. ErwinKreyszig,AdvancedEngineeringMathematics,JohnWiley&Sons,Inc., 10th Edition, 2016.

REFERENCE BOOKS

1. BaliN.P,GoyalM,WatkinsC,AdvancedEngineeringMathematics,Laxmi Publications Pvt. Limited, 2007.
2. JamesStewart,Calculus:EarlyTranscendental,CengageLearning,NewDelhi, 7th Edition, 2013.
3. Narayanan,S and Manicavachagom Pillai,T.K.,Calculus Volume I and II, S. Viswanathan (Printers & Publishers), Pvt., Ltd., 1997.
4. Howard Anton, Irl C. Bivens, StephenDavis,Calculus Early Transcendentals, John Wiley & Sons, Inc., 11th Edition, 2016.
5. Srimanta Pal and Subodh C. Bhunia, Engineering Mathematics, Oxford University Press, 2015.

COURSE CODE	COURSE TITLE	L	T	P	C
UPH1176	ENGINEERING PHYSICS	3	0	0	3

OBJECTIVES

- Comprehend and identify different crystal structures and their imperfections.
- Explain the elastic and thermal properties of materials and understand their significance.
- Develop an understanding of quantum mechanical phenomena and their applications.
- Provide an overview of the characteristics of sound, architectural acoustics and the production, detection and applications of ultrasound.
- Explain the origin of laser action, production of laser, fiber optics and their applications.

UNIT I

Single crystalline, polycrystalline and amorphous materials– single crystals - Lattice – Unit cell– Bravais lattice – Lattice planes – Miller indices – d spacing in cubic lattice – Calculation of number of atoms per unit cell – Atomic radius – Coordination number – Packing factor for SC, BCC, FCC and HCP structures – Diamond and graphite structures (qualitative treatment) - Crystal Imperfections – Point, line (Edge and Screw dislocations – Burger vectors) Surface (stacking faults) and Volume defects.

UNIT II

Properties of matter: Elasticity- Hooke's law - Relationship between three moduli of elasticity– stress -strain diagram– Poisson's ratio –Factors affecting elasticity– Torsional stress & deformations – Twisting couple – Torsion pendulum - theory and experiment–bending of beams-bending moment–cantilever: theory and experiment–uniform and non-uniform bending: theory and experiment-I-shaped girdersThermal Physics: Modes of heat transfer – thermal conduction, convection and radiation - thermal conductivity-Linear heat flow (Derivation) – Lee's disc method – Radial heat flow – Rubber tube method – conduction through compound media (series and parallel) – Formation of ice on ponds.

UNIT III

Acoustics: Classification and characteristics of Sound - decibel - Weber–Fechner law – Sabine's formula - derivation using growth and decay method —factors affecting acoustics of buildings and their remedies - Types ofAcoustic absorbers - Methods of determination of Absorption Coefficient.

Ultrasonics: Production of ultrasonic by Magnetostriction and piezoelectric methods - acoustic grating -Non-Destructive Testing –pulse echosystem through transmission and reflection modes - A, B and C – scan displays.

UNIT IV

Black body radiation – Planck's theory (derivation) – Deduction of Wien's displacement law and Rayleigh – Jeans' Law from Planck's theory – Compton Effect. Theory and experimental verification – Properties of Matter waves – wave particle duality - Schrödinger's wave equation– Time independent and time dependent equations – Physical significance of wave function – Particle in a one-dimensional boxand extension to three-dimensional box – Degeneracy of electron energy states - Scanning electron microscope - Transmission electron microscope.

UNIT V

Photonics: Spontaneous and stimulated emission- Population inversion -Einstein's A and B coefficients –Conditions for Laser action - Types of lasers – Nd: YAG, CO2,Diode lasers- Industrial and Medical Applications. Fibre optics: Principle and propagation of light in optical fibres – Numerical aperture and Acceptance angle - Types of optical fibres (material, refractive index, mode) –Losses in fibers - attenuation, dispersion, bending - Fibre Optical Communicationsystem (Block diagram) - Active and passive fibre sensors- pressure and displacement.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Analyze crystal structures and the influence of imperfections on their properties.
- Demonstrateandexplainthegeneralconceptsofelasticandthermalpropertiesof materials.
- Explain quantum mechanical theories to correlate with experimental results and their applications to material diagnostics.
- Analyzetheapplicationsofacousticsandultrasonictoengineeringandmedical disciplines.
- Elucidate the principle and working of lasers and optical fibers, and their applications in the field of industry, medicine and telecommunication.

TEXTBOOKS

1. Gaur, R.K., and Gupta, S.L., Engineering Physics, Dhanpat Rai Publishers, 2012.
2. Serway, R.A. & Jewett, J.W., Physics for Scientists and Engineers, Cengage Learning, 2010.

REFERENCE BOOKS

1. Halliday, D., Resnick, R. & Walker, J. Principles of Physics. Wiley, 2015.
2. Tipler, P.A. &Mosca, G. Physics for Scientists and Engineers with Modern Physics, WH Freeman, 2007.
3. Avadhanulu M. N., Kshirsagar P. G, A textbook of Engineering Physics, S. Chand & Co. Ltd., Ninth Revised Edition, 2012.

COURSE CODE	COURSE TITLE	L	T	P	C
UCY1176	ENGINEERING CHEMISTRY	3	0	0	3

OBJECTIVES

- To make the students conversant with boiler feed water requirements, related problems and water treatment techniques.
- To give an overview about Polymers
- To develop an understanding of the basic concepts of phase rule and its application
- To make the students conversant with the types of fuels, calorific value calculations, manufacture of solid, liquid and gaseous fuels.
- Toprovideknowledgeonelectrochemicalcell,measurementofredoxpotential, electrochemical corrosion and control, electroplating.

UNIT I WATER AND ITS TREATMENT 9

Hardness of water – types – expression of hardness – units – estimation of hardness of water by EDTA – numerical problems – Alkalinity- boiler troubles (scale and sludge) – treatment of boiler feed water–Internal treatment(phosphate, colloidal, sodium aluminate and calgon conditioning) external treatment–Ion exchange process, zeolite process –desalination of brackish water- Reverse Osmosis.

UNIT II POLYMET CHEMISTRY 9

Introduction: Classification of polymers – Natural and synthetic; Thermoplastic and Thermosetting. Functionality – Degree of polymerization. Types and mechanism of polymerization: Addition (Free Radical, cationic and anionic); condensation and copolymerization. Properties of polymers: Tg, Tacticity, Molecular weight – weight average, number average and polydispersity index Preparation, properties and uses of PVC, PE, PS Nylon 6.6, and Epoxy resin. Bio-degradable polymers. Effect of polymers on environment.

UNIT III PHASE RULE AND ALLOYS 9

Phase rule:Introduction, definition of terms with examples, one component system -water system - reduced phase rule - thermal analysis and cooling curves - two component systems - lead-silver system - Pattinson process – magnesium-zinc system. Alloys: Introduction- Definition- properties of alloys- significance of alloying – heat treatment of steel.

UNIT IV

Fuels: Introduction - classification of fuels - coal - analysis of coal (proximate and ultimate) - carbonization - manufacture of metallurgical coke (Otto Hoffmann method) - petroleum - manufacture of synthetic petrol (Bergius process) - knocking - octane number - diesel oil - cetane number - natural gas - compressed natural gas (CNG) - liquefied petroleum gases (LPG) - power alcohol and biodiesel.

Combustionoffuels:Introduction-calorificvalue-higherandlowercalorificvalues- theoretical calculation of calorific value – theoretical air for combustion (problems) - flue gas analysis (ORSAT Method).

UNIT V ELECTORCHEMISTRY AND CORROSION 10

Electrochemicalcell-redoxreaction,electrodepotential-originofelectrodepotential- oxidation potential - reduction potential, measurement and applications – electrochemical series and its significance - Nernst equation (derivation and problems). Corrosion – causes – factors - types chemical, electrochemical corrosion (galvanic, differential aeration), corrosion control – material selection and design aspects – electrochemical protection – sacrificial anode method and impressed current cathodic method. Paints - constituents and function. Electroplating of Copper and electrode less plating of nickel.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- The knowledge gained on water treatment techniques,Polymers, Phase rule, Fuels and electrochemistry and Corrosion will facilitate better understanding of engineeringprocesses and applications for further learning.

TEXTBOOKS

1. Jain P.C. and Monika Jain, Engineering Chemistry Dhanpat Rai Publishing Company (P) Ltd, New Delhi, 2015.
2. Vairam, S., Kalyani, P. and Suba Ramesh, Engineering Chemistry, Wiley India Pvt, Ltd, New Delhi, 2013.

REFERENCE BOOKS

1. Dara, S.S and Umare, S.S., A Textbook of Engineering Chemistry, S. Chand & Company Ltd, New Delhi, 2015.
2. Friedrich Emich, Engineering Chemistry, Scientific International Pvt, Ltd, New Delhi, 2014.
3. Prasanta Rath, Engineering Chemistry, Cengage Learning India Pvt, Ltd, Delhi, 2015.
4. ShikhaAgarwal,EngineeringChemistry-FundamentalsandApplications,Cambridge University Press, Delhi, 2015.

COURSE CODE	COURSE TITLE	L	T	P	C
UGE1176	PROBLEM SOLVING AND PROGRAMMING IN PYTHON	3	0	0	3

OBJECTIVES

- To solve algorithmic problems
- To abstract and specify problems
- To compose programs in Python using iteration and recursion
- To construct programs in Python using functions

UNIT I ALGORITHMIC PROBLEM SOLVING 9

Algorithms, building blocks of algorithms (statements, state, control flow, functions); Notation (pseudo code, flow chart, programming language); specification, composition, decomposition, iteration, recursion.

UNIT II DATA, EXPRESSION, STATEMENT, CONDITIONAL 9

Data and types: int, float, boolean, string, list; variables, expressions,statements, simultaneous assignment, precedence of operators; comments; in-builtmodules and functions; Conditional: boolean values and operators, conditional (if), alternative (if-else), case analysis (if-elif-else).

UNIT III ITERATION, FUNCTION, STRINGS 9

Iteration: while, for, break, continue, pass; Functions: function definition, function call, flow of execution, parameters and arguments, return values, local and global scope, recursion; Strings: string slices, immutability, string functions and methods, string module.

UNIV IV LISTS, TUPLES 9

Lists: list operations, list slices, list methods, list loop, mutability, aliasing, cloning lists, list parameters, nested lists, list comprehension; Tuples: tuple assignment, tuple as return value, tuple operations.

UNIV V DICTIONARIES, FILES 9

Dictionaries: operations and methods, looping and dictionaries, reverse lookup, dictionaries and lists; Files: Text files, reading and writing files, format operator, file names and paths; command line arguments.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Think logically to solve programming problems and write solutions in pseudo code.
- Understand and develop simple Python programs using conditionals and loops.
- Decompose a program into functions.
- Represent compound data using Python lists, tuples, dictionaries.
- Perform input/output with files.

TEXTBOOKS

1. Allen B. Downey, Think Python: How to Think Like a Computer Scientist, 2nd Edition, Updated for Python 3, Shroff/O'Reilly Publishers, 2016.(http://greenteapress.com/wp/think-python/)
2. Guido van Rossum and Fred L. Drake Jr, An Introduction to Python – Revised and updated for Python 3.2, Network Theory Ltd., 2011.

REFERENCE BOOKS

1. John V Guttag, Introduction to Computation and Programming Using Python, Revised and expanded Edition, MIT Press, 2013.
2. Robert Sedgewick, Kevin Wayne, Robert Dondero, Introduction to Programming in Python: An Inter-disciplinary Approach, Pearson India Education Services Pvt. Ltd., 2016.
3. Timothy A. Budd, Exploring Python, McGraw Hill Education (India) Private Ltd., 2015.
4. Kenneth A. Lambert, Fundamentals of Python: First Programs, CENGAGE Learning, 2012.
5. Charles Dierbach, Introduction to Computer Science using Python: A Computational Problem-Solving Focus, Wiley India Edition, 2013.
6. PaulGries,JenniferCampbellandJasonMontojo,PracticalProgramming:An Introduction to Computer Science using Python 3, Second edition, Pragmatic Programmers, LLC, 2013.

COURSE CODE	COURSE TITLE	L	T	P	C
UGE1176	ENGINEERING GRAPHICS	1	0	4	3

OBJECTIVES

- Todevelopthegraphicskillsforcommunicationofconcepts,ideasanddesignof engineering products.
- To expose them to existing national standards related to technical drawings.

UNIT I PLANE CURVES AND FREEHAND SKETCHING 10

Basic Geometrical constructions, Curves used in engineering practices: Conics – Construction of ellipse, parabola and hyperbola by eccentricity method – Drawing of tangents and normal to the above curves.Visualization concepts and Free Hand sketching: Visualization principles – Representation of Three-Dimensional objects – Layout of views- Freehand sketching of multiple views from pictorial views of objects

UNIT II PROJECTION OF POINTS, LINES AND PLANE SURFACES 15

Orthographic projection principles - Principal planes - First angle projection - Layout of views - Projection of points. Projection of straight lines (only First angle projections) inclined to both the principal planes - Determination of true lengths and true inclinations by rotating line method and traces. Projection of planes (polygonal and circular surfaces) inclined to both the principal planes by rotating object method.

UNIT III PROJECT OF SOLIDS 15

Projection of simple solids like prisms, pyramids, cylinder, cone and truncated solids when the axis is inclined to one of the principal planes by rotating object method.

UNIT IV PROJECTION OF SECTIONED SOLIDS AND DEVELOPMENT OF SURFACES 20

Sectioning of above solids in simple vertical position when the cutting plane is inclined to theone of the principal planes and perpendicular to the other – obtaining true shape of section. Development of lateral surfaces of truncated solids (simple position only) – Prisms, pyramids cylinders and cones.

UNIV V ISOMETRIC PROJECTION AND BUILDING DRAWING 15

Principles of isometric projection – isometric scale – Isometric projections of simple solids and truncated solids - Prisms, pyramids, cylinders, cones- combination of two solid objects in simple vertical positions – Building drawing – Plan, Elevation and Sectional View showing Foundation of simple buildings like pump room.

TOTAL PERIODS: 75

OUTCOMES

On successful completion of this course, the student will be able to

- Familiarize with the fundamentals and standards of Engineering graphics
- Perform freehand sketching of basic geometrical constructions and multiple views of objects.
- Project orthographic projections of lines and plane surfaces.
- Draw projections and solids and development of surfaces.
- Visualize and to project isometric and perspective sections of simple solids. Read a building drawing.

TEXTBOOKS

1. Natarajan,K.V.,ATextbookofEngineeringGraphics,DhanalakshmiPublishers, Chennai, 2009.
2. Venugopal, K. and Prabhu Raja, V., Engineering Graphics, New Age International (P) Limited, 2008.

REFERENCE BOOKS

1. Bhatt, N.D., and Panchal, V.M., Engineering Drawing, Charotar Publishing House, 50th Edition, 2010.
2. BasantAgarwal,andAgarwal,C.M.,EngineeringDrawing,TataMcGrawHill Publishing Company Limited, New Delhi, 2008.
3. Gopalakrishna, K.R., Engineering Drawing (Vol. I &II Combined), Subhas Stores, Bangalore, 2007.
4. Luzzader J Warren, and Jon M Duff, Fundamentals of Engineering Drawing with an introduction to Interactive Computer Graphics for Design and Production, Prentice Hall of India Pvt. Ltd, New Delhi, Eastern Economy Edition, 2005.
5. Parthasarathy, N.S., and Vela Murali, Engineering Graphics, Oxford University, Press, New Delhi, 2015.
6. Shah M.B., and Rana B.C., Engineering Drawing, Pearson, 2nd Edition, 2009.
7. Bhattacharyya, B., and Bera, S.C., Engineering Graphics, I.K. International Publishing House Pvt. Ltd., New Delhi.

Publication of Bureau of Indian Standards:

1. IS 10711 – 2001: Technical products Documentation – Size and lay out of drawing sheets.
2. IS 9609 (Parts 0 & 1) – 2001: Technical products Documentation – Lettering.
3. IS 10714 (Part 20) – 2001 & SP 46 – 2003: Lines for technical drawings.
4. IS 11669 – 1986 & SP 46 – 2003: Dimensioning of Technical Drawings.
5. IS 15021 (Parts 1 to 4) – 2001: Technical drawings – Projection Methods.

Special points applicable to End Semester Examinations on Engineering Graphics:

- There will be five questions, each of either-or type covering all units of the syllabus.
- All questions will carry equal marks of 20 each making a total of 100.
- The answer paper shall consist of drawing sheets of A3 size only. The students will be permitted to use appropriate scale to fit solution within A3 size.
- The examination will be conducted in appropriate sessions on the same day.

COURSE CODE	COURSE TITLE	L	T	P	C
UGE1197	PROGRAMMING IN PYTHON LAB	0	0	3	1.5

OBJECTIVES

- To solve problems using algorithms and flowcharts
- To write, test, and debug simple Python programs.
- To develop and execute programs using Python programming constructs.

SUGGESTIVE EXERCISES

1. Use Linux shell commands, use Python in interactive mode, and an editor
2. Writesimple programs (areaofageometric shape, simple interest, solve quadratic equation, net salary).
3. Write programs using conditional statements (leap year, maximum of 2 numbers, maximum of 3 numbers, simple calculator, grade of the total mark).
4. Develop programs using loops and nested loops (gcd, prime number, integer division, sum of digits of an integer, multiplication table, sum of a series, print patterns, square root using Newton's method).
5. Develop programs using function (sine and cosine series, Pythagorean triplets).
6. Develop programs using recursion (efficient power of a number, factorial, Fibonacci number).
7. Develop programs using strings (palindrome, finding substring) without using in-built functions.
8. Develop programs using list and tuples (linear search, binary search, selection sort, insertion sort, quick sort).
9. Develop programs using nested lists (matrix manipulations).
10. Develop simple programs using dictionaries (frequency histogram, nested dictionary).
11. Develop programs using Files (read and write files).
12. Develop programs to perform any task by reading arguments from command line.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- To write, test, and debug simple Python programs.
- To implement Python programs with conditionals and loops.
- Use functions for structuring Python programs.
- Represent compound data using Python lists, tuples, and dictionaries.
- Read and write data from/to files in Python.

COURSE CODE	COURSE TITLE	L	T	P	C
UGS1197	PHYSICS AND CHEMISTRY LAB	0	0	3	1.5

OBJECTIVES

- The Properties of Matter
- The Optical properties like Interference and Diffraction.
- Optical Fibre Characteristics
- Characteristics of Lasers.
- Electrical & Thermal properties of Materials
- and enable the students to enhance accuracy in experimental measurements.

LIST OF EXPERIMENTS

(A minimum of 8 experiments to be performed from the given list)

- Determination of Young's modulus of the material of the given beam by Non-uniform bending method.
- Determination of rigidity modulus of the material of the given wire using torsion pendulum.
- Determination of velocity of sound in the given liquid and compressibility of the liquid using
- Ultrasonic interferometer.
- Determination of wavelength of mercury spectra using Spectrometer and grating.
- Determination of dispersive power of prism using Spectrometer.
- Determination of grating element/wavelength, and particle size/ wavelength using a laser.
- Determination of Numerical and acceptance angle of an optical fiber.
- Determination of thickness of a thin wire using interference fringes.
- Determination of the coefficient of viscosity of the given liquid using Poiseuille's method.
- Determination of energy band gap of the semiconductor.
- Determination of coefficient of thermal conductivity of the given bad conductor using Lee's disc.
- Determination of specific resistance of the material of the given wire using Carey Foster's bridge

OUTCOMES

On successful completion of this course, the student will be able to

- demonstrate the ability to formulate, conduct, analyze and interpret the results of experiments related to study/determination of
- The physical properties of materials like elasticity, compressibility, and viscosity.
- Theopticalproperties ofmaterialssuchasdiffraction, interference andNumerical aperture.
- Thermal and electrical properties of materials such as conductivity and band gap.

CHEMISTRY LABORATORY
(A minimum of 8 experiments to be performed from the given list)

The chemistry laboratory course consists of experiments illustrating the principles of chemistry relevant to the study of science and engineering.

OBJECTIVES

- Understand and apply the basic techniques involved in quantitative analysis
- Apply the knowledge gained in theory course

LIST OF EXPERIMENTS

1. Estimation of Hardness by EDTA method
2. Estimation of Chloride in water
3. Estimation of Alkalinity of water
4. Estimation of iron by spectrophotometry
5. Determination of the strength of strong acid by pH metry
6. Determination of the strength of strong acid by conductometry
7. Determination of the strength of mixture of strong and weak acids by conductometry
8. Estimation of Na by flame photometry
9. Estimation of Fe2+ by potentiometric titration
10. DeterminationofDegreeofPolymerizationofalowMolecularweightwatersoluble polymer
11. Determination of rate of corrosion of mild steel in acidic medium
12. Estimation of Barium chloride by conductometry titration

OUTCOMES

On successful completion of this course, the student will be able to

- Evaluate the quality of water
- Determine the metals and ions present in any given sample using various analytical techniques
- Measure properties such as conductance of solutions, redox potentials

TOTAL PERIODS: 45

COURSE CODE	COURSE TITLE	L	T	P	C
UEN1276	TECHNICAL ENGLISH	3	0	0	3

OBJECTIVES

- To develop strategies and skills to enhance their ability to read and comprehend texts in engineering and technology.
- To improve their ability to write convincing job applications and effective reports.
- To develop their speaking skills to make technical presentations, participate in group discussions.
- To strengthen their listening skill which will help them, comprehend lectures and talks in their areas of specialization

UNIT I INTRODUCING TECHNICAL WRITING 9

Reading: Reading short technical texts from journals, newspapers and checking their comprehensionWriting: Definitions, instructions, recommendations, checklist.
Language Development: Subject Verb Agreement, numerical adjectives. Vocabulary Development: Avoidance of jargon, Technical vocabulary Speaking: Asking for and giving DirectionsListening: Listening to technical talks with comprehension tasks.

UNIT II INTERPRETING CHARTS AND GRAPHS 9

Reading: Practice in chunking and speed reading.
Writing: Interpreting charts, graphs and other kinds of visual information. Language Development: Use of passive voice in technical writing
Vocabulary Development: Important Latin and other foreign expressions in use. Speaking: Talking about Processes (Technical and General)
Listening:Listening Comprehension of a discussion on a technical topic of common interest by three or four participants (real life as well as online videos).

UNIT III PREPARING FOR A PRESENTATION 9

Reading: Reading longer texts for detailed understanding. (GRE/IELTS practice tests) Writing: Describing general or technical processes using appropriate flow charts. Vocabulary Development: Informal vocabulary and formal substitutes (based on a small grammatically streamlined sample)
LanguageDevelopment:EmbeddedsentencesandEllipsis(allowedanddisallowed types)
Speaking: 5-minute presentations on technical/general topics
Listening: Listening Comprehension (IELTS practice tests)

UNIT IV WRITING AND SPEAKING IN FORMAL SITUTATION 9

Reading: Technical reports, advertisements and minutes of meeting Writing: Writing minutes of a meeting, reports and general essays Vocabulary Development: paraphrasing, analogy, collocationsLanguage Development: if conditionals and other kinds of complex sentencesSpeaking: Public Speaking (debates, extempore, just a minute)
Listening: Listening to eminent voices of one's choice (in or outside the class, followed by a discussion in the class)

UNIT V WRITING REPORTS 9

Reading: Extensive Reading (short stories, novels, poetry and others) Writing: reports (accident, issue-/survey-based), minutes of a meeting Vocabulary Development: Archaisms and contemporary synonyms, clichés. Language Development: Summarizing, Elaboration.
Speaking: Talk to public personalities and share the experience in class.
Listening: Extensive Listening. (radio plays, rendering of poems, audio books and others)

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Apply reading strategies to comprehend technical texts and write areaspecifictexts effortlessly.
- Listen and comprehend lectures and talks in science and technology.
- Speak appropriately and effectively in varied formal and informal contexts.
- Write technical documents like reports, emails, resume and winning job applications.

TEXTBOOKS

1. Board of editors, Fluency in English: A Course book for Engineering and Technology, Orient Blackswan, Hyderabad, 2016.

REFERENCE BOOKS

1. Sudharshana, N.P., and Saveetha, C., English for Technical Communication, Cambridge University Press, New Delhi, 2016.
2. Raman,Meenakshi,Sharma,andSangeetha,TechnicalCommunicationPrinciplesand Practice, Oxford University Press, New Delhi, 2014.
3. Kumar, Suresh, E., Engineering English, Orient Blackswan, Hyderabad, 2015.
4. Booth L Diana, Project Work, Oxford University Press, 2014.
5. Grussendorf, Marion, English for Presentations, Oxford University Press, 2007.
6. Means, L. Thomas and Elaine Langlois, English & Communication for Colleges, Cengage Learning, USA, 2007.

COURSE CODE	COURSE TITLE	L	T	P	C
UMA1276	COMPLEX FUNCTIONS AND LAPLACE TRANSFORMS	3	2	0	4

OBJECTIVES

- Understand C-R equations and use it in the construction of Analytic Functions.
- Understand the methods of Complex Integration using Cauchy's Integral Formula and Cauchy Residue theorem, finding Taylor's and Laurent's Series expansions.
- Find the Laplace Transforms of standard Functions.
- Find the Inverse Laplace Transform of a function and use it in solving Differential Equations.
- Understand the concept of Divergence and curl and use it in evaluating Line, Surface and Volume integrals.

UNIT I ANALYTIC FUNCTIONS 12

Analyticfunctions–necessaryandsufficientconditions–Cauchy-Riemannequationsin Cartesian and polar form (with proof) - Properties-harmonic functions, Construction of analytic function, conformal mapping, some standard transformations -$w = z + c, cz, \frac{1}{z}, z^2$ bilinear transformation.

UNIT II COMPLEX INTEGRATION 12

Line integral - Cauchy's integral theorem – Cauchy's integral formula – Taylor's and Laurent's series – Singularities – Residues – Residue theorem – Application of residue theorem for evaluation of real integrals – Use of circular contour and semicircular contour (except the poles on the real axis).

UNIT III LAPLACE TRANSFORMS 12

Definition, properties, existence conditions – Transforms of elementary functions – Transform of unit step function and unit impulse function – Shifting theorems – Transforms of derivatives and integrals – Initial and final value theorems, Evaluation of integrals by Laplace transforms, periodic functions, Inverse transforms – Convolution theorem

UNIT IV ORDINARY DIFFERENTIAL EQUATIONS 12

Solutionofsecondandhigherorderlineardifferentialequationwithconstantcoefficients ($f(x) = e^{mx}, sinmx, cosmx, x^n$), Solving linear second order ordinary differential equations with constant coefficients using Laplace transforms, Simultaneous linear equations with constant coefficients of first order.

UNIT V VECTOR CALCULUS 12

Gradient and directional derivative – Divergence and curl –Vector identities – Irrotational and Solenoidal vector fields, Line integral over a plane curve, Surface integral -Area of a curved surface, Volume integral,Green's, Gauss divergence and Stoke's theorems – Verification and application in evaluating line, surface and volume integrals.

TOTAL PERIODS: 60

OUTCOMES

On successful completion of this course, the student will be able to

- Solve problems in Analytic functions and construction of analytic functions using C-R equations.
- Evaluate problems using Cauchy's integral formula and Cauchy residue theorem and find Taylor's and Laurent's series expansion of a given function.
- Obtain the Laplace Transforms of standard functions.
- Solve Differential Equations of Second order and Simultaneous linear equations with constant coefficients of first order using Laplace Transform.
- Solveproblemsusingdivergenceandcurlandevaluateline,SurfaceandVolume integrals.

TEXTBOOKS

1. Grewal, B.S., Higher Engineering Mathematics, 43rd Edition, Khanna Publishers, 2016.
2. Erwin Kreyszig, Advanced Engineering Mathematics, 10th Edition, John Wiley & Sons, Inc., 2016.

REFERENCE BOOKS

1. Bali,N.P.,Goyal,M.,Watkins,C.,AdvancedEngineeringMathematics,Laxmi Publications Pvt. Limited, 2007.
2. Boyce,W.E.,andDiPrima,R.C.,ElementaryDifferentialEquationsandBoundary Value Problems, Wiley India, 2012.
3. GeorgeB.ThomasJr.,MauriceD.Weir,JoelR.Hass,Thomas'Calculus:Early Transcendental, 13th Edition, Pearson Education, 2013.
4. O'Neil. P. V., Advanced Engineering Mathematics, 7th Edition, Cengage Learning India Pvt., Ltd, New Delhi, 2011.
5. Howard Anton, Irl C. Bivens, StephenDavis,CalculusEarlyTranscendentals,11th Edition, John Wiley & Sons, Inc., 2016.
6. Srimanta Pal and Subodh C. Bhunia, Engineering Mathematics, Oxford University Press, 2015.
7. Srivastava, A.C., and Srivastava, P.K., Engineering Mathematics Volume I and II, PHI learning Pvt. Ltd, 2011.

COURSE CODE	COURSE TITLE	L	T	P	C
UPH1276	PHYSICS FOR INFORMATION SCIENCE	3	0	0	3

OBJECTIVES

- Understand the transport properties of conducting materials and their modelling using classical and quantum theories.
- Analyzethephysicsofsemiconductorsandrelatetheirmicroscopicpropertiesto observable bulk phenomena.
- Understand the origin of magnetism and data storage principles.
- Study the fundamentals of optical materials and their applications to display devices. Develop an overview of Nano materials and their applications to Nano devices.

UNIT I CONDUCTING MATERIALS 9

Classification of solids- Conductors – classical free electron theory of metals – Electrical and thermal conductivity – Wiedemann – Franz law – Lorentz number – Draw backs of classical theory – Quantum free electron theory – Density of energy states - Fermi distribution function – Effectoftemperature onFermiFunction — carrierconcentration inmetals–Electronin periodic potential – Bloch theorem – Kronig - Penney model (qualitative) – Band theory of solids (qualitative), tight binding approximation, E-k curves and effective mass

UNIT II SEMICONDUCTING MATERIALS 9

Intrinsic semiconductor – Bond and energy band diagrams–Concept of hole - carrier concentration derivation – Fermi level – Variation of Fermi level with temperature – electrical conductivity – band gap determination – extrinsic semiconductors – Bond and energy band diagrams - carrier concentration derivation in n-type and p-type semiconductor – variation of Fermi level with temperature and impurity concentration – Direct and indirect band semiconductors –Velocity –electricfieldrelations-Halleffect–Determination ofHall coefficient – Applications& Devices – Formation of PN junction –energy band diagram - biased and unbiased conditions.

UNIT III DATA STORAGE PRINCIPLES 9

Originofmagneticmoment–Bohrmagneton,atomicmagneticmoments–magnetic permeability and susceptibility – Microscopic and macroscopic classification of magnetic materials – comparison of Dia and para magnetism and Ferro magnetism – Ferromagnetism: originandexchangeinteraction-saturationmagnetizationandCurietemperature-Domain theory – Hysteresis (based on domain theory) – soft and hard magnetic materials – Magnetic principles in computer data storage – Magnetic hard disc – GMR Sensor- Principle of GMR- Parts of a magnetic hard disc - CD-ROM-WORM- Magneto-optical storage, recording and reading systems - Holographic optical data storage.

UNIT IV OPTICAL MATERIALS AND DISPLAY DEVICES 9

Absorption emissionandscatteringoflightinmetals,insulatorsandsemiconductors(conceptsonly) - Carrier generation and recombination in semiconductors –-LED – OLED- Semiconductor Laser diodes (Homo and double hetero junction) –Photo detectors – Photodiodes and Photoconductors(concepts only) – Solar cell – Liquid crystal display - Charged Coupled Devices

UNIT V NANO DEVICES 9

Nanomaterials–Properties,Applications,Sizeeffect-Density ofstatesinquantumwell, quantum wire and quantum dot structures- Quantum confinement- Quantum well and Quantum dotlasers-Franz-Keldysheffect-QuantumConfinedStarkeffect–QuantumWellElectroAbsorption modulators- Magnetic semiconductors – Spintronics.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Estimate the conducting properties of materials based on classical and quantum theories and understand the formation of energy band structures.
- Acquireknowledgeonbasicsofsemiconductorphysicsanditsapplicationto PN junction devices.
- Gainknowledge onmagneticpropertiesofmaterialsandtheirapplicationstodata storage.
- Relate the use of optical materials to display devices.
- Understandquantummechanicsofnanostructuresandtheirapplicationto Nano electronics and Spintronics.

TEXTBOOKS

1. Adaptation by Balasubramanian, R, Callister's Material Science and Engineering, Wiley India Pvt. Ltd., 2nd Edition, 2014.
2. Kasap,S.O.,PrinciplesofElectronicMaterialsandDevices,(SpecialIndianEdition) McGraw-Hill Education, 3rd Edition, 2017.

REFERENCE BOOKS

1. Umesh K Mishra &Jasprit Singh, Semiconductor Device Physics and Design, Springer, 2008.
2. Pallab Bhattacharya, Semiconductor Optoelectronic Devices, Pearson, 2nd Edition, 2017.
3. Wahab, M.A. Solid State Physics: Structure and Properties of Materials, Narosa Publishing House, 2009.
4. Gaur, R.K. & Gupta, S.L., Engineering Physics, Dhanpat Rai Publishers, 2012.
5. Salivahanan, S.,Rajalakshmi, A., Karthie, S., Rajesh, N.P.,Physics for Electronics Engineering and Information Science, McGraw Hill Education (India) Private Limited, 2018.
6. Avadhanulu, M.N.,P. G. Shirsagar, A textbook of Engineering Physics, S. Chand & Co. Ltd. Ninth Revised Edition, 2012.
7. Theuwissen, A.J., Solid state imaging with Charge-Coupled Devices, Kluwer-Academic Publisher, Springer 1995.

COURSE CODE	COURSE TITLE	L	T	P	C
UCT1276	ENVIRONMENTAL SCIENCE	3	0	0	3

OBJECTIVES

- To understand the structure and functions of the ecosystems and biodiversity among life forms within an ecosystem
- To realize the importance of various natural resources and its sustainable use
- To address the various environmental issues related to various types of pollution.
- To address various social issues and the role of various environmental machineries to ensure proper environmental regulations
- To understand the influence of human population on environment issues and role of IT as a tool to minimize the environmental problems

UNIT I ENVIRONMENT, ECOSYSTEMS AND BIODIVERSITY 9

Definition, scope and importance of environment– concept, structure and function of an ecosystem – energy flow- food chains, food webs and ecological pyramids – ecological succession Introduction to biodiversity definition and types– values of biodiversity- India as a mega-diversity nation–hot-spotsofbiodiversity –threatstobiodiversity-endangered and endemic species ofIndia -conservation of biodiversity: In-situ andex-situ conservation of biodiversity.

UNIT II NATURAL RESOURCES 9

Uses, over-exploitation of natural resources: Forest, Water, Mineral, Food, Energy and Land. Case studies on over exploitation of natural resources -Role of an individual in conservation of natural resources- Equitable use of resources for sustainable lifestyles.

UNIT III ENVIRONMENTAL POLLUTION 9

Definition – causes, effects and control measures of: (a) Air pollution (b) Water pollution (c) Soil pollution (d) Marine pollution (e) Noise pollution (f) Thermal pollution (g) Nuclear hazards – solid waste management: causes, effects and control measures of municipal solid wastes – role of an individual in prevention of pollution – pollution case studies – disaster management: floods, earth quake, cyclone and landslides.

UNIT IV SOCIAL ISSUES AND ENVIRONEMENT 9

From unsustainable to sustainable development – water conservation, rain water harvesting, watershed management – role of non-governmental organization - Social Issues and possible solutions – climate change, global warming, acid rain, ozone layer depletion, case studies – environment protection act – Air (Prevention and Control of Pollution) act – Water (Prevention and control of Pollution) act – Wildlife protection act – Forest conservation act – enforcement machinery involved in environmental legislation- central and state pollution control boards- Public awareness.<Contents for Unit IV>

UNIT V HUMAR POPULATION AND THE ENVIRONMENT 9

Population growth, variation among nations – population explosion – family welfare programme –environment and human health – human rights – value education –HIV/AIDS-women and child welfare – role of information technology in environment and human health – Case studies.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Understand the importance of the environment and describe the structure and functions of an ecosystem.
- Identify the value and need for conservation of biodiversity.
- Know the importance of natural resources and its equitable use for sustainable lifestyles. Explain the causes, effects and control measures of different types of pollution. Understand various environmentally related social issues and their solutions.
- Recall the tools for environmental regulations
- Relate the role of environment in human population growth and development Get knowledge about various techniques used for environmental monitoring and management.

TEXTBOOKS

1. Anubha Kaushik, and Kaushik, C. P., Environmental Science and Engineering, New Age International Publishers, 14thEdition, 2014.
2. Benny Joseph, Environmental Science and Engineering, Tata McGraw-Hill, New Delhi, 2006.

REFERENCE BOOKS

1. Gilbert M. Masters, Introduction to Environmental Engineering and Science, 2nd Edition, Pearson Education, 2004.
2. Tyler Miller, G., and Scott E. Spoolman, Environmental Science, Cengage Learning India Pvt, Ltd, Delhi, 2014.

COURSE CODE	COURSE TITLE	L	T	P	C
UEE1276	BASIC ELECTRICAL, ELECTRONICS AND MEASUREMENT ENGINEERING	3	1	0	3.5

OBJECTIVES

- To understand the fundamentals of electronic circuit constructions.
- To learn the fundamental laws, theorems of electrical circuits and also to analyze them
- To study the basic principles of electrical machines and their performance
- To study the different energy sources, protective devices and their field applications
- To understand the principles and operation of measuring instruments and transducers

UNIT I ELECTRICAL CIRCUIT ANALYSIS 12

Ohms Law, Kirchhoff's Law-Instantaneous power- series and parallelcircuit analysis with resistive, capacitive and inductive network - nodal analysis, mesh analysis- network theorems – Thevenin's theorem,Norton theorem,maximumpower transfer theoremandsuperposition theorem, three phase supply-Instantaneous, Reactive and apparent power-star delta conversion.

UNIT II ELECTRICAL MACHINES 12

DC and AC Rotating Machines: Types, Construction, principle, EMF and torque equation, application Speed Control- Basics of Stepper Motor – Brushless DC motors- Transformers-Introduction- types and construction, working principle of Ideal transformer- EMF equation- All day efficiency calculation.

UNIT III UTILIZATION OF ELECTRICAL POWER 12

Renewable energy sources-wind and solar panels. Illumination by lamps- Sodium Vapour, Mercuryvapour, Fluorescent tube. Domestic refrigerator and air conditioner-Electric circuit, construction and working principle. Batteries-NiCd, Pb Acid and Li ion–Charge and Discharge Characteristics.Protection-needforearthing,fusesandcircuitbreakers.EnergyTariff calculation for domestic loads.

UNIT IV ELECTRONIC CIRCUITS 12

PN Junction-VI Characteristics of Diode, Zener diode, Transistors configurations - amplifiers. Op amps- Amplifiers, oscillator, rectifiers, differentiator, integrator, ADC, DAC. Multivibrator using 555 Timer IC. Voltage regulator IC using LM 723, LM 317.

UNIT V ELECTRICAL MEASUREMENT 12

Characteristic of measurement-errors in measurement, torque in indicating instruments- moving coil and moving iron meters, Energy meter and watt meter. Transducers- classification-thermo electric, RTD, Strain gauge, LVDT, LDR and piezoelectric. Oscilloscope-CRO.

TOTAL PERIODS: 60

OUTCOMES

On successful completion of this course, the student will be able to

- Upon completion of this course, the students will be able to
- Understand the essentials of electric circuits and analysis.
- Understand the basic operation of electric machines and transformers Introduction of renewable sources and common domestic loads.
- Introduction to measurement and metering for electric circuits.

TEXTBOOKS

1. Kotharti,D.P.,andNagarath,I,J.,BasicElectricalandElectronicsEngineering, 3rd Edition, McGraw Hill, 2016.
2. Sukhija, M.S., and Nagsarkar, T.K., Basic Electrical and Electronic Engineering, Oxford, 2016.

REFERENCE BOOKS

1. LalSeksena,S.B.,andKaustuvDasgupta,FundamentsofElectricalEngineering, Cambridge, 2016.
2. Theraja, B.L., Fundamentals of Electrical Engineering and Electronics. S. Chand & Co, 2008.
3. Sahdev, S.K., Basic of Electrical Engineering, Pearson, 2015.
4. John Bird, Electrical and Electronic Principles and Technology, Fourth Edition, Elsevier, 2010.
5. Mittle, Mittal, Basic Electrical Engineering, 2nd Edition, Tata McGraw-Hill, 2016.
6. Wadhwa, C.L., Generation, Distribution and Utilization of Electrical Energy, New Age International Pvt. Ltd., 2003.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1201	PROGRAMMING IN C	3	1	0	3.5

OBJECTIVES

- To solve programming problems in C
- To use basic programming concepts of C
- To develop programs using loops, functions, arrays, pointers, strings, structures and files

UNIT I BASICS 12

Algorithm -- structure of a C program -- data types: built-in, user-defined -- variables and constants -- input and output statements -- operators, expressions, and assignment statements -- control statements.

UNIT II FUNCTIONS 12

Function prototype -- function definition and call -- passing parameters -- built-in functions-- Scope: local, external, static, blocks-- recursion; Pre-processor directives.

UNIT III ARRAYS AND POINTERS 12

Arrays: declaration, initialization -- one dimensional arrays – multi-dimensional arrays; Pointers: pointer operators -- pointer operations -- array of pointers, pointer to a function -- passing arrays and pointers to functions.

UNIT IV STRINGS, STRUCTURES AND UNIONS 12

Strings: constants and variables, reading and writing strings, string operations, string library; Command line arguments; Structures: nested structures -- structures and functions -- array of structures-- pointer to a structure – unions – type definition (typedef); Dynamic memory allocation.

UNIT V FILES 12

Files:openingandclosingadatafile--creatingadatafile--processingadatafile-- unformatted data files -- formatted input and output -- line input and output – file error handling; Standard library functions.

TOTAL PERIODS: 60

OUTCOMES

On successful completion of this course, the student will be able to

- Write pseudo-code for solving programming problems.
- Represent data with suitable structured data types.
- Develop C programs using appropriate programming constructs.
- Organize programs into functions and files.

TEXTBOOKS

1. ByronGottfried,ProgrammingwithC(Schaum'sOutlinesSeries),McGraw-Hill Education, 3rd Edition, 2017.
2. Brian W Kernighan and Dennis M Ritchie, The C Programming Language, Pearson Education India, 2nd Edition, 2015.

REFERENCE BOOKS

1. Reema Thareja, Programming in C, Oxford University Press, 2nd Edition, 2016.
2. Yashwant Kanetkar, Let Us C, BPB Publications, 14th Edition, 2016.
3. King,K.N.,CProgrammingAModernApproach,W.W.Norton&Company, 2nd Edition, 2008.

COURSE CODE	COURSE TITLE	L	T	P	C
UGE1297	DESIGN THINKING AND ENGINEERING PRACTICES LAB	0	0	3	1.5

OBJECTIVES

- To provide exposure to the students with hands on experience on various basic engineering practices in Civil, Mechanical, Electrical and Electronics Engineering.

GROUP A (CIVIL & MECHANICAL)

I - CIVIL ENGINEERING PRACTICE

Buildings:

(a) Study of plumbing and carpentry components of residential and industrial buildings. Safety aspects.

Plumbing Works:

(a) Study of pipeline joints, its location and functions: valves, taps, couplings, unions, reducers, and elbows in household fittings.

(b) Preparation of plumbing line sketches for water supply and sewage works.

(c) Hands-on-exercise:Basicpipeconnections–Mixedpipematerialconnection–Pipe connections with different joining components.

Carpentry using Power Tools only:

(a) Study of the joints in roofs, doors, windows and furniture.

(b) Hands-on-exercise: Woodwork, joints by sawing, planning and cutting.

(c) Wood working - Demonstration of wood working machinery and furniture manufacturing.

II - MECHANICAL ENGINEERING PRACTICE

Basic Machining: (a) Drilling Practice Sheet Metal Work:

(a) Forming & Bending: (b) Model making – Trays, dustpan and funnels. (c) Different type ofjoints.

Machine assembly practice:

(a) Study of centrifugal pump (b) Study of air conditioner

Design Thinking: Students will be trained to dismantle, understand the functional / aesthetic aspects of the product and to assemble the following components like (a) Three jaw Chuck Assembly (b) Iron Box (c) Pedestal Fan (d) Lathe Tailstock.

Demonstration on:

(a) Smithy operations, upsetting, swaging, setting down and bending. Example – Exercise – Production of hexagonal headed bolt. (b) Foundry operations like mould preparation for gear and step cone pulley. (c) Fitting – Exercises – Preparation of square fitting and V – fitting models. (d) Arc welding and Gas Welding (e) Lathe operations.

GROUP B (ELECTRICAL & ELECTRONICS)

III ELECTRICAL ENGINEERING PRACTICE

1. Residential house wiring using switches, fuse, indicator, lamp and energy meter.
2. Fluorescent lamp wiring.
3. Staircase wiring
4. Measurement of electrical quantities – voltage, current, power & power factor in RLC circuit.
5. Measurement of energy using single phase energy meter.
6. Measurement of resistance to earth of electrical equipment.

IV ELECTRONICS ENGINEERING PRACTICE

1. Study of Electronic components and equipment – Resistor color coding measurement of AC
2. signal parameter (peak-peak, RMS period, frequency) using CR.
3. Study of logic gates AND, OR, EX-OR and NOT.
4. Generation of Clock Signal.
5. Soldering practice – Components Devices and Circuits – Using general purpose PCB.
6. Measurement of ripple factor of HWR and FWR.

TOTAL PERIODS: 45

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1297	PROGRAMMING IN C LAB	0	0	3	1.5

OBJECTIVES

- To develop, implement, and execute C programs.
- To use various programming constructs and features in C
- To understand the concept of modular programming using functions

SUGGESTIVE EXERCISES

1. Programs using, I/O statements and conditional constructs (if, switch, ternary)
2. Programs using looping constructs (for, while, do-while)
3. Programs using functions with different parameter passing techniques: Call by value (e.g. swapping two numbers), call by reference (e.g. changing the elements of an array), Recursion (e.g. binary search)
4. Programs using one dimensional array (e.g. inserting an element after every i^{th} positionin an array, Insertion sort)
5. Programs using multi-dimensional arrays (e.g. matrix manipulations)
6. Programs using strings and their operations (e.g. concatenation of strings, extracting a substring, checking for palindrome, search for a given string using binary search)
7. Programs to demonstrate pointers to functions (e.g. simple arithmetic calculator)
8. Programs to demonstrate simple structure manipulations (e.g. generating a transcript with CGPA and class obtained)
9. Programs to pass structures to functions (e.g. operations on complex numbers, difference between times)
10. Programs to pass array of structures to a function (e.g. generate invites to N parents for a meeting by passing the details of students)
11. Programs to demonstrate file operations (e.g. count the number of characters, words and lines in a file, replace a specific word with the given word in the same file)

MINI PROJECTS

Develop an application modularly using C programming constructs. (Eg. Library management system, Online ticket reservation system – Train/Bus/Airways, Hotel reservation system)

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Implement C programs, compile, debug, and execute.
- ImplementprogramsinCwithcorrectdatarepresentationandappropriateprogramming constructs.
- Compare arrays and pointers in C.
- Use structures and files in C.
- Develop programs modularly and refine incrementally.

COURSE CODE	COURSE TITLE	L	T	P	C
UMA1377	DISCRETE MATHMATICS	3	2	0	4

OBJECTIVES

- To understand the classical logic, implications and equivalences, normal forms and its applications.
- TolearnMathematicalInduction.
- Tolearnthebasicconceptsingraphtheoryandprove simpleproperties.
- Tounderstandthebasicsofgrouptheory.
- TounderstandthebasicsoflatticesandBooleanalgebra.

UNIT I LOGIC AND PROOFS 13

Propositional Logic–Propositional equivalences–Predicates and quantifiers– Nested quantifiers–Rules of inference–Introduction to proofs–Proof methods and strategy–Normal forms–Applications to switching circuits.

UNIT II COMBINATORICS 11

Mathematical induction–Strong induction–The pigeon hole principle–Recurrence relations–Solving linear recurrence relations using generating functions –Inclusion and Exclusion Principle and its applications.

UNIV III GRAPHS 10

Graphs–Graph terminology and special types of graphs–Subgraphs–Matrix representation of graphs and graph isomorphism–Connectivity–Eulerian and Hamilton graphs.

UNIT IV ALGEBRAIC STRUCTURES 13

Algebraic systems–Semi groups and monoids–Groups–Subgroups Homomorphisms–Normal subgroup and coset–Lagrange's theorem–Definitions and examples of Rings and Fields.

UNIT V LATTICES AND BOOLEAN ALGEBRA 13

Partial ordering–Posets–Lattices as Posets–Properties of lattices–Lattices as algebraic systems–Sublattices–Direct product and Homomorphism; Boolean algebra –Stone's representation Theorem.

TOTAL PERIODS: 60

OUTCOMES

On successful completion of this course, the student will be able to

- Writesimple proofs using PropositionalandFirstOrderLogics
- Writesimple proofs using mathematicalinduction
- Provesimple graphproperties.
- Explainbasicconceptsingrouptheorysuchassemigroups,monoidsand groups.
- Solveproblemsinpartial orderingrelations,equivalencerelationsandlattices.

TEXTBOOKS

1. KennethHRosen,“DiscreteMathematicsand itsApplications”,7th Edition, SpecialIndianedition,TataMcGraw Hill,NewDelhi,2017.
2. TremblayJPandManoharR,“DiscreteMathematicalStructureswithAp plicationstoComputerScience”, 30thReprint,Tata McGraw Hill, New Delhi,2011.

REFERENCE BOOKS

1. Ralph PGrimaldi,“Discreteand CombinatorialMathematics: An AppliedIntroduction”,4thEdition, PearsonEducationAsia,2007.
2. Thomas Koshy,“DiscreteMathematicswith Applications”,ElsevierPublications, 2006.
3. SeymourLipschutz,MarkLipson,“DiscreteMathematics”,Schaum’sOutlines,3rdEditi on, Tata McGraw Hill,2010.
4. CLLiu,DP Mohapatra,“ElementsofDiscreteMathematics”,4thEdition, McGraw Higher Education,2012.
5. JohnMHarris, JeffryLHirst, MichaelJ Mossinghoff, “Combinatorics and Graph Theory”,VerlagNewYork,2008.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1301	DIGITAL PRINCIPLES AND SYSTEM DESIGN	3	0	0	3

OBJECTIVES

- To design digitalcircuitsusingsimplified Boolean functions
- To analyze and design combinational circuits
- To analyze and design synchronous and asynchronous sequential circuits
- To understand Programmable Logic Devices
- To write HDL code for combinational and sequential circuits.

UNIT I BOOLEAN ALGEBRA AND LOGIC GATES 9

Number Systems– Arithmetic Operations – BinaryCodes–BooleanAlgebra and Logic Gates – Theorems and Properties of Boolean Algebra – Boolean Functions– Canonical andStandard Forms– Simplification of BooleanFunctions usingKarnaugh Map – Logic Gates – NANDand NOR Implementations.

UNIT II COMBINATIONAL LOGIC 9

Combinational Circuits– Analysis andDesignProcedures – BinaryAdder– Subtractor – DecimalAdder – BinaryMultiplier – Magnitude Comparator – Decoders – Encoders – Multiplexers – Introduction to HDL – HDL Modelsof Combinational circuits.

UNIT III MEMORY AND PROGRAMMABLE LOGIC 9

RAM– Memory Decoding – ErrorDetection andCorrection – ROM– Programmable Logic Array – Programmable Array Logic – Sequential Programmable Devices.

UNIT IV SYNCHRONOUS SEQUENTIAL LOGIC 9

Sequential Circuits – Storage Elements:Latches, Flip–Flops; Analysis of Clocked Sequential Circuits– StateReduction andAssignment – DesignProcedure – Registersand Counters – HDL Modelsof Sequential Circuits.

UNIT V ASYNCHRONOUS SEQUENTIAL LOGIC 9

Analysis and Designof Asynchronous Sequential Circuits– Reduction of State and Flow Tables – Race free State Assignment – Hazards.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Simplify Boolean functions usingKMap (K3)
- Analyze and design Combinational Circuits(K4)
- Analyze and design Sequential Circuits(K4)
- Understand various memories andimplement Booleanfunctions using Programmable Logic Devices (K3)
- Write HDL code for combinational and Sequential Circuits(K3).

TEXTBOOKS

1. MorrisR Mano, MichaelD Ciletti, "Digital Design:With an Introduction to the Verilog HDL, VHDL, and System Verilog", 6th Edition,Pearson Education, 2017.
2. S Salivahanan, S Arivazhagan, "Digital Circuit sand Design", 5th Edition, OxfordUniversity Press, 2018.

REFERENCE BOOKS

1. G K Kharate, "Digital Electronics", Oxford University Press, 2010.
2. JohnF Wakerly, "Digital Design Principles and Practices", 5thEdition, Pearson Education, 2017.
3. Charles H Roth Jr, LarryL Kinney,"Fundamentals of Logic Design", 6th Edition,CENGAGE Learning, 2013.
4. Donald D Givone, "Digital Principles and Design", Tata McGraw Hill, 2003.
5. Thomas L Floyd, "Digital Fundamentals", 11th Edition, Pearson Education, 2017.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1302	DATA STRUCTURES	3	0	0	3

OBJECTIVES

- To understand the concepts of ADT
- To comprehend the notion of linearand nonlineardatastructures
- To design suitable datastructures for different problems andimplement themin a high
- level programming language
- To understand the different typesof searching and sorting techniques
- To learn the introductory concepts of hashing techniques.

UNIT I LINEAR DATA STRUCTURES – LIST 10

Algorithm Analysis: Running time calculations; Abstract Data Types (ADTs); List ADT: Array implementation of lists – Linked lists – Circular linked lists – Doubly linked lists – Applications of Lists: Polynomial manipulation.

UNIT II LINEAR DATA STRUCTURES – STACKS AND QUEUES 8

Stack ADT: Stack model – Implementation of stacks – Applications: Balancing symbols – Infix to postfix conversion – Evaluating postfix expressions; Queue ADT: Queue model – Array implementation of queues – Applications of queues.

UNIT III NONLINEAR DATA STRUCTURES – TREES 9

Preliminaries: Implementation of trees – Tree traversals with an application; Binary trees: Expression trees; Binary search tree ADT; AVL trees; BTrees; Priority Queues: Binary heap.

UNIT IV NONLINEAR DATA STRUCTURES – GRAPHCS 9

Graph Algorithms: Definitions – Representation of graphs; Graph Traversals: Breadth first traversal – Depth first traversal; Topological sort – Shortest Path Algorithms: Dijkstra's algorithm – All pairs shortest path.

UNIT V SEARCHING, SORTING AND HASHING TECHNIQUES 9

Searching: Linear search – Binary search; Sorting: Selection sort – Shell sort; Hashing: Hash function – Separate chaining – Open addressing – Rehashing – Extendible hashing.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Elucidate and implement various operations of List (K3)
- Develop applications usingstack and queue (K3)
- Implement the operations of balanced and unbalanced trees (K3)
- Develop applications using shortest path and traversal algorithms of graph(K3)
- Choosesuitable searching andsorting algorithms to solve various computing problems (K4).

TEXTBOOKS

1. M A Weiss, “DataStructures andAlgorithm Analysis in C”, 2nd Edition, Pearson Education, 2002.
2. RichardF Gilberg, Behrouz A Frouzan, “Data Structures: A Pseudocode Approach with C”, 2nd Edition,Cengage India,2007.

REFERENCE BOOKS

1. A V Aho,J E Hopcroft, J D Ullman,“DataStructures andAlgorithms”, Pearson Education, 1st EditionReprint, 2003.
2. R F Gilberg,B A Forouzan, “Data Structures”, 2nd Edition, Thomson India Edition,2005.
3. Ellis Horowitz, Sartaj Sahni, Susan Anderson Freed, “Fundamentals of Data Structures in C”, 2nd Edition,University Press, 2008.
4. S Sridhar, “Designand Analysis of Algorithms “, 1st Edition,OxfordUniversityPress, 2014.
5. Byron Gottfried, Jitender Chhabra, “Programming with C” (Schaum’s Outline Series), 3rd Edition,McGrawHill Higher Education, 2010.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1303	OBJECT ORIENTED PROGRAMMING USING JAVA	3	0	0	3

OBJECTIVES

- To learn the basics of Object-Oriented Programming
- To knowthe principles of inheritance and polymorphism
- To learn the concepts of genericmethods and genericcollections.

UNIT I INTRODUCTION 10

Principles of OOP: Classes – Objects – Data hiding – Data encapsulation – Inheritance – Polymorphism; Definition of Classes: Objects – Methods – Access specifiers – Static and final classes and members; Object Construction and Destruction – Fundamental programming structures in Java; Streams:Input Output – Examples in Java.

UNIT II INHERITANCE 10

Inheritance: Definition – Types of inheritance: Single – Multilevel – Multiple – Hierarchical – Hybrid; Subclassconstructors – Interfaces in Java:Definition – Implementation – Extending interfaces – Inheritance versus delegation – Inheritance rules – Innerclasses – Examples in Java.

UNIT III POLYMORPHISM AND EXCEPTION HANDLING 7

Polymorphism: Method overloading andoverriding – Dynamic method dis patch;Exceptions: Hierarchy – Built-inexceptions – Creating ownexception; Packagesin Java – Examples in Java.

UNIT IV GENERIC TYPES AND METHODS 9

Definition andconcepts: Genericclasses andgenericmethods – Generictypes– Restrictions and limitations – Inheritance rules for generictypes – Reflections– Examples in Java.

UNIT V GENERIC COLLECTIONS FOR ADTS AND ALGORITHMS 9

Introduction to collections – Collection Classes and Interfaces:Array list – Linked list – Queue – Set – Trees;Iterators for collections – Mapclass – Algorithms: Sorting– Searching – User defined algorithms – Examples in Java.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Solve problems usingclasses and objects (K3)
- Develop programs usinginheritance and interfaces (K3)
- Applythe concepts of polymorphism (K3)
- Designapplications usinggenericprogramming (K3)
- Develop algorithms usinggenericcollections (K3).

TEXTBOOKS

1. Danny Poo, Derek Kiong, Swarnalatha Ashok, “Object Oriented Programming and Java”, 2nd Edition,Springer Publication, 2008.

2. Herbert Schildt,“Java:The Complete Reference”, 8th Edition,McGrawHill Education, 2011.

REFERENCE BOOKS

1. Timothy Budd, “Understanding Object oriented programming with Java”, Updated Edition,Pearson Education, 2000.
2. C ThomasWu,“Anintroduction to Object oriented programming withJava”, 4th Edition,Tata McGraw-Hill Publishing company Ltd., 2006.
3. CayS Horstmann, GaryCornell, “CoreJava Volume–I Fundamentals”,9th Edition, Prentice Hall, 2013.
4. PaulDeitel, Harvey Deitel, “JavaSE 8 for programmers”, 3rdEdition, Pearson, 2015.
5. Steven Holzner, “Java 2 Black book”, Dreamtech press, 2011.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1304	UNIX AND SHELL PROGRAMMING	2	0	2	3

OBJECTIVES

- To learn UNIX Commands
- To learn vi Editor
- To learn File handling in UNIX
- To learn Shell Programming.

UNIT I FILE SYSTEM 6

Computer system – UNIX environment – UNIX structure; File Systems – Security and File Permission; vi Editor: Modes and commands.

UNIT II COMMANDS 6

Filters; Regular Expressions; grep; sed: Addresses, Commands; awk: Patterns, Actions, Associative Arrays.

UNIT III BASH BASICS 6

Introduction to Shells; customizing your environment; Basic Shell Programming: Shell scripts, shell variables, string operators, command substitution.

UNIT IV SHELL PROGRAMMING 6

Flow control; Command-line options and Typed Variables: Integer variables and arithmetic, Arrays.

UNIT V INPUT / OUTPUT 6

Input/Output and Command-Line Processing: I/O redirectors, string I/O, Command Line Processing; Process Handling; Job control, signals, subshells; bash Administration.

TOTAL PERIODS(THEORY): 30

LIST OF EXPERIMENTS

1. Exercises on basic UNIX commands.
2. Exercises on file and directory handling.
3. Exercises on security and file permissions.
4. Exercises on pipes,quotes, aliases and variables.
5. Exercises on filters.
6. Exercises on sed
7. Exercises on awk.

TOTAL PERIODS(PRACTICAL): 30
TOTAL PERIODS: 60

OUTCOMES

On successful completion of this course, the student will be able to

- Work in UNIX environment using command line (K2)
- Handle files system proficiently (K2)

- Use vi editor with proficiency (K3)
- Work with the bash shell (K2)
- Automate with shell scripts (K3)

TEXTBOOKS

1. Behrouz Forouzan, Richard Gilberg "UNIX and Shell Programming: A Textbook", Thomson Learning, 2005 (Units I & II).
2. Cameron Newham, Bill Rosenblatt "Learning the Bash Shell", Third Edition, O'Reilly, 2005 (Units III, IV & V).

REFERENCE BOOKS

1. Mike Loukides, Tim O'Reilly, Jerry Peek, Shelley Powers, "Unix Power Tools", 3rd Edition, O'Reilly, 2009.
2. Sumitabha Das, "Unix: Concepts and Applications", Tata McGraw Hill,2017.
3. Richard Blum, Christine Bresnahan, "Linux Command Line and Shell Scripting Bible", Wiley, 2015.
4. Stephen Kochan, Patrick Wood, "Shell Programming in UNIX, Linux and OSX", Addison Wesley, 2016.
5. Randal K Michael, "Mastering Unix Shell Scripting", 2nd Edition, Wiley, 2008.

COURSE CODE	COURSE TITLE	L	T	P	C
UEC1351	PRINCIPLES OF COMMUNICATION ENGINEERING	3	0	0	3

OBJECTIVES

- To study the various analogand digitalmodulation techniques
- To study the principles behind information theory and coding
- To study the various digitalcommunication techniques
- To understand the conceptof spread spectrum system.

UNIT I ANALOG MODULATION 9

Amplitude Modulation – AM, DSBSC, SSBSC, VSB – Spectrum, modulators and demodulators – Angle modulation – PM and FM – Spectrum, FM modulation – Direct andIndirect methods, FM Demodulation – Foster Seeley Discriminator, PLL as FM demodulator, Superheterodyne receivers.

UNIT II PULSE MODULATION 9

Low pass sampling theorem – Quantization – PAMLine coding – PCM, DPCM, DM, andADM – Time DivisionMultiplexing, Frequency DivisionMultiplexing.

UNIT III DIGITAL MODULATION AND TRANSMISSION 9

ASK, FSKPSK: BPSK, QPSK & DPSK – QAM – Comparison – Principles of Mary signaling, ISI – Pulse shaping – Eye pattern.

UNIT IV INFORMATION THEORY AND CODING 9

Entropy – Source codingtheorem – ShannonFano coding,Huffman Coding – Channel capacity– ShannonHartley law – Error controlcodes – Cyclic codes – Encoder,Syndrome Calculator – Convolution Coding – Viterbi Decoding.

UNIT V SPREAD SPECTRUM AND MULTIPLE ACCESS 9

PN sequences – properties – msequence – DSSS – Processing gain,Jamming margin – FHSS – Synchronization andtracking – Multiple Access– FDMA, TDMA, CDMA.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Apply analog and digital communication techniques (K3)
- Design analog and digital communication systems (K3)
- Use pulse communication techniques & application (K3)
- Analyze Source and Error control coding (K3)
- Analyze the performance of spread spectrum systems (K3).

TEXTBOOKS

1. Wayne Tomasi, “Electronic communications systems: Fundamentals through
1. Advanced”, 5th edition, Pearson PrenticeHall.
2. S Haykin, “Digital Communications”, John Wiley, 2005.

REFERENCE BOOKS

1. B P Lathi,"Modern DigitalandAnalogCommunication Systems", 3rd edition, OxfordUniversity Press, 2007.
2. H P Hsu, Schaum Outline Series, "Analog and Digital Communications", TMH, 2006.
3. B Sklar, "Digital Communications Fundamentals and Applications", 2/e Pearson Education, 2007.
4. H Taub,D L Schilling,G Saha, "Principles of Communication Systems",3/e, TMH, 2007.
5. J G Proakis, "Digital Communication", 4th Edition, Tata McGraw Hill Company, 2001.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1311	DIGITAL DESIGN LAB	0	0	3	1.5

OBJECTIVES

- To understand the various basic logic gates
- To design and implement the various combinational circuits
- To design and implement combinational circuitsusingMSI devices.
- To design and implement sequential circuits
- To understand and code with HDL programming

SUGGESTIVE LAB EXPERIMENTS

1. Verificationof Boolean Theorems usingbasic gates.
2. Design and implementation of combinational circuits using basic gates for arbitrary functions, code converters.
3. Designand implement half/full adder and subtractor.
4. Designand implement combinational circuitsusingMSI devices:
    a. 4bit binary adder/subtractor
    b. Paritygenerator/Checker
    c. Magnitude Comparator
    d. Application usingmultiplexers
5. Designand implement shift registers.
6. Designand implement synchronous counters.
7. Designand implement asynchronous counters.
8. Coding combinational circuitsusingHDL.
9. Coding sequential circuitsusingHDL.
10. Designand implementation of a simpledigitalsystem (Mini Project).

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to:

- Understand basic gates and universal gates to design a digitalcircuit (K3)
- DesignandImplement various combinational circuitsusinglogic gates(K3)
- DesignandImplement various combinational circuitsusingMSI devices(K3)
- Designand Implement various sequential circuits(K3)
- Simulate various combinational and sequential circuitsusingHDL (K3).

LABORATORY REQUIREMENT FOR BATCH OF 25 STUDENTS

Hardware:

1. Digital trainer kits25
2. Digital ICs required for the experiments in enoughnumbers

Software:

1. HDL simulator.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1312	DATA STRUCTURES LAB	0	0	4	2

OBJECTIVES

- Understand and apply suitable datastructures, linearand nonlinear
- Design the datastructure for the given problem and implement them in a high-level programming language
- Understand theconcepts anddifferences in searching andsorting techniques
- Learn the introductory concepts of hashing techniques.

SUGGESTIVE EXPERIMENTS

1. Arrayimplementation of list ADT (e.g. Manage student records)
2. Represent polynomial expression as linkedlist andwritefunctions for polynomial manipulation
3. Implement doubly linkedlist ADT (e.g. Insert,delete,search,merge,re verse operations)
4. Arrayimplementation of Stack ADT (e.g. Balancingparenthesis, infix to postfix conversion, evaluation of postfix expression)
5. Arrayimplementation of Queue ADT (e.g. Simulating printer jobs)
6. Implement an expression tree. Produce its preorder, inorder, andpost order traversals
7. Implement binary searchtree
8. Implement AVL trees (e.g. Implement dictionary)
9. Implement priority queue using binary heaps (e.g. Storing employee records basedupon salary)
10. Graph representation and implement graph traversal algorithms
11. Implement Dijkstra'salgorithm usinggraph (e.g. Findthe shortest route to connectone city to another)
12. Implement binarysearchand any two sorting algorithms
13. Implement hashing usingseparate chaining technique

TOTAL PERIODS: 60

OUTCOMES

On successful completion of this course, the student will be able to:

- Implement linear and nonlinear data structures to solve a given problem(K3)
- Implement nonlinear data structures and use them for various applications (K3)
- Analyze a problem statement and suggest an appropriate linear/ nonlinear data structure for solving it (K4)
- Implement sorting, searching and hashing algorithms (K2)
- Implement hashing technique for data storage and retrieval (K3).

LABORATORY REQUIREMENT FOR BATCH OF 25 STUDENTS

Hardware:

1. Standalone Systems: 25 Nos

Software:

1. C / C++ Compiler

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1313	OBJECT ORIENTED PROGRAMMING USING JAVA LAB	0	0	3	1.5

OBJECTIVES

- To buildsoftware development skills usingobject-oriented programming for real world applications
- To develop applications usinginheritance and polymorphism
- To understand and implement genericprogramming.

SUGGESTIVE EXPERIMENTS

1. Program usingclasses and objects (Eg: Electricity bill generation)
2. Program to perform stringoperations usingArrayList.
3. Program to implement packages (Eg: Currency converter)
4. Program usingInheritance (Eg: Payrollapplication)
5. Program to implement Polymorphism (E.g. Area of different shapes)
6. Program to implement genericmethods (Eg: Sorting)
7. Designa Java interface for ADTs. (E.g.Stack, Queue)
8. Program to perform file operations. (E.g.existence, permission, type, length in bytes)
9. Develop a mini project for any application usingthe constructs of Java.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to:

- Develop applications in Java usingclasses and methods(K3)
- Applyinheritance and interface concepts to write programs(K3)
- Write Java programs usingpolymorphic methods and objects(K3)
- Develop applications using generic methods and generic programming(K3)
- Apply appropriate features of object-oriented programming paradigm to design mini project(K3).

LABORATORY REQUIREMENT FOR BATCH OF 25 STUDENTS

Hardware:

1. Standalone Systems: 25 Nos

Software:

1. Java

COURSE CODE	COURSE TITLE	L	T	P	C
UMA1478	PROBABILITY AND STATISTICS	3	2	0	4

OBJECTIVES

- To identify the standard distributions and apply them in solving problems
- To understand the concept of two-dimensional random variables and solve problems in finding the Joint probabilities and correlation between them
- To perform hypothesis testing using normal, t-distribution and F-distribution
- To evaluate the tests of significance in analysis of variance
- To calculate the various statistical quality control measurements.

UNIT I RAMDAM VARIABLES 12

Probability – Axioms of probability— Conditional probability – Bayes' theorem– Discrete and continuous random variables – Moments – Moment generating functions – Binomial, Poisson, Geometric, Uniform, Exponential, Gamma, Nor- mal distributions – Functions of a random variable.

UNIT II TWO-DIMENSIONAL RANDOM VARIABLES 12

Joint distributions – Marginal and conditional distributions – Covariance – Correlation and linearregression – Transformation of random variables – Central limit theorem (for independent and identically distributed random variables).

UNIT III TESTS OF SIGNIFICANCE 12

Sampling distributions – Small andlargesample test – Test basedon Normal and *t* distribution (Single and difference of meanand proportion)– $\chi 2$– Test for goodness of fit, Independence of attributes- F test for variance.

UNIT IV DESIGN OF EXPERIMENTS 12

Completely randomized design – Randomized block design – Latin square de- sign – Factorialdesign.

UNIT V STATISTICAL QUALITY COTROL 12

Control charts for measurements (X and R charts) – Control charts for attributes(p, c and npcharts) – Tolerance limits – Acceptance sampling.

TOTAL PERIODS: 60

OUTCOMES

On successful completion of this course, the student will be able to:

- Identify standard distributions and apply them
- Solve problems in two-dimension random variables and find the correlation between them
- Identify andapply the suitable testing of hypothesis under normal and t and F distribution
- Solve problems in analysis of variance
- Analyze quality control by applying control chart methods.

TEXTBOOKS

1. Milton J S and Arnold J C, “Introduction to Probability and Statistics”, 4thEdition,3rd Reprint, Tata McGrawHill, 2008.
2. JohnsonR A and Gupta C B, “Miller and Freund’s Probability and Statistics for Engineers”, 8th Edition,Pearson Education, 2011.

REFERENCE BOOKS

1. DevoreJ L, “Probability andStatisticsfor Engineering andthe Sciences”,7th Edition,ThomsonBrooks/Cole, International Student Edition,2008.
2. WalpoleR E, MyersR H, MyersS L, Ye K, “Probability andStatisticsforEngineers and Scientists”, 8th Edition,Pearson Education, Asia, 2007.
3. Ross S M, “Introduction to Probability andStatisticsfor Engineers andScientists”, 3rd Edition,Elsevier, 2004.
4. Spiegel M R, Schiller J, Srinivasan R A, “Schaum’s Outline of TheoryandProblems of Probability and Statistics”, Tata McGrawHill, 2004.
5. Gupta S C, KapoorV K, “Fundamentals of Mathematical Statistics”, Sul- tan and Chand Company, 2009.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1401	COMPUTER ORGANIZATION AND ARCHITECTURE	3	0	0	3

OBJECTIVES

- To learn the basic structure and operations of a computer
- To learnthe arithmetic andlogic unitandimplementation of fixed-point and floating-point arithmetic unit
- To learn the basics of pipelined execution
- To understand the memory hierarchies, cache andvirtual memories and communication with I/O devices
- To understand parallelism and multi-core processors.

UNIT I BASIC STRUCTURE OF A COMPUTER SYSTEM 9

Functional Units -- Basic Operational Concepts -- Performance; Instructions: Language of the computer -- Operations, Operands -- Instruction representation; Logical operations -- Decision making; MIPS addressing.

UNIT II ARITHMETIC FOR COMPUTER 9

Addition and subtraction; Multiplication; Division; Floating Point Representation: Floating point operations; Sub word parallelism.

UNIT III PROCESSOR AND CONTROL UNIT 9

A Basic MIPS implementation: Building a data path – Control implementation scheme; Pipelining: Pipelined data path and control -- Handling data hazards & Control hazards -- Exceptions -- Issues in predictive branching: Spectre and Meltdown.

UNIT IV MEMORY & I/O SYSTEMS 9

Memory Hierarchy; Memory technologies; Cache Memory: Measuring and improving cache performance; Virtual Memory: TLBs; Accessing I/O devices -- Interrupts; Direct memory access; Bus structure – Bus operation -- Arbitration; Interface circuits; USB.

UNIT V PARALLEL PROCESSORS 9

Parallel processing challenges; Flynn's classification: SISD – MIMD -- SIMD -- SPMD and Vector Architectures; Hardware multithreading; Multi-core processors and other shared memory multiprocessors; Introduction to Graphics Processing Units.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to:

- Understand the basics structure of computers, operations and instructions(K2)
- Designarithmetic and logic unit (K3)
- Understand pipelined execution and design controlunit (K3)
- Designof various memory systems andunderstand I/O communication(K3)
- Understand parallel processing architectures (K2).

TEXTBOOKS

1. DavidA Patterson, John L Hennessy, “Computer Organization andDesign: The Hardware/Software Interface”, 5th Edition,Morgan Kaufmann/ Elsevier, 2014 (Units I, III, IV, V).
2. Carl Hamacher, ZvonkoVranesic, SafwatZaky, NaraigManjikian, “Computer Organization andEmbedded Systems”, 6th Edition,Tata McGraw Hill, 2012 (Unit II).

REFERENCE BOOKS

1. William Stallings, “Computer Organization and Architecture – Designing for Performance”, 8th Edition,Pearson Education, 2010.
2. John P Hayes,“Computer Architecture andOrganization”, 3rdEdition, Tata McGrawHill, 2012.
3. John L Hennessey, David A Patterson, “Architecture – A Quantitative Approach”, 5th edition, Morgan Kaufmann / Elsevier, 2012 (Units I, III).
4. MorrisManoM, “Computer SystemArchitecture”, Revised3rd Edition, Pearson Publication, 2017.
5. Chakraborty P, “Computer Architecture andOrganization”, JAICO PublishingHouse, 2010.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1402	OPERATING SYSTEMS	3	0	0	3

OBJECTIVES

- Understand the structure and functions of OS
- Learn about Processes, Threads and Scheduling algorithms
- Understand the principles of concurrency and Deadlocks
- Learn various memory management schemes
- Study I/O management and File systems.

UNIT I OPERATING SYSTEMS OVERVIEW 9

Computer System Overview: Basic elements -- Instruction execution – Interrupts -- Memory hierarchy -- Cache memory -- Direct memory access -- Multiprocessor and multicore organization; Operating System Overview: Objectives and functions -- Evolution of operating system;Computer system organization; Operating System Structure and Operations: System calls -- System programs -- OS generation and system boot.

UNIT II PROCESS MANAGEMENT 12

Processes: Process concept -- Process scheduling -- Operations on processes -- Interprocess communication; Threads: Overview -- Multithreading models -- Thread issues; Process synchronization -- Critical section problem -- Mutex locks -- Semaphores -- Monitors; CPU Scheduling: FCFS, SJF, Priority, Round robin, Rate Monotonic and EDF scheduling; Deadlocks -- Avoidance -- Prevention -- Detection and Recovery.

UNIT III MEMORY MANAGEMENT 8

Main Memory: Contiguous memory allocation -- Segmentation -- Paging -- 32 and 64bit architecture Examples; Virtual Memory: Demand paging -- Page replacement algorithms -- Allocation -- Thrashing.

UNIT IV STORAGE MANAGEMENT 9

Mass Storage Structure: Overview -- Disk scheduling and management; File System Storage: File concepts -- Directory and disk structure -- Sharing and protection; File System Implementation: File system structure -- Directory structure -- Allocation methods -- Free space management -- I/O systems.

UNIT V CASE STUDY 7

Linux Vs Windows: Design principles -- Process management -- Scheduling -- Memory management -- File systems; Mobile OS: iOS and Android -- Introduction and architecture.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to:

- Describe the basic services and functionalities of operating systems (K2)
- Analyze various scheduling algorithms, and understand the different deadlock, prevention and avoidance schemes (K4)

- Understand the different memory management schemes (K2)
- Understand the functionality of file systems (K2)
- Compare and contrast Linux, Windows and mobile operating systems (K3).

TEXTBOOKS

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, "Operating SystemConcepts", 9th Edition,John Wiley and Sons Inc, 2012.
2. Neil Smyth, "iPhone iOS 4 Development Essentials - XCode", 4th Edition, Payload media,2011.

REFERENCE BOOKS

1. Ramaz Elmasri, A Gil Carrick, David Levine, "Operating Systems A Spiral Approach", Tata McGraw-Hill Edition, 2010.
2. Achyut S Godbole, Atul Kahate, "Operating Systems", McGraw-Hill Education, 2016.
3. Andrew S Tanenbaum, "Modern Operating Systems", 2nd Edition, Pearson Education, 2004.
4. Harvey M Deitel, "Operating Systems", 3rd Edition, Pearson Education, 2004.
5. Daniel P Bovet, Marco Cesati, "Understanding the Linux Kernel", 3rd Edition, O'Reilly,2005.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1403	DESIGN AND ANALYSIS OF ALGORITHMS	3	0	2	4

OBJECTIVES

- Learn the algorithm analysis techniques
- Become familiar with the different algorithm design techniques
- Understand the limitations of algorithm power.

UNIT I INTRODUCTION AND ANALYSIS 10

Introduction: Fundamentals of algorithmic Problem solving -- Important problem types; Fundamentals of the Analysis of Algorithm Efficiency: Analysis framework -- Asymptotic notations and basic efficiency classes -- Mathematical analysis for recursive and non-recursive algorithms.

UNIT II BRUTE FORCE AND DIVIDE AND CONQUER 8

Brute Force: String matching -- Closest-pair problem; Exhaustive search: Traveling salesman problem -- Knapsack problem.Divide and Conquer: Merge sort -- Quicksort -- Multiplication of large integers and Strassen's matrix multiplication.

UNIT III DYNAMIC PROGRAMING AND GREEDY TECHIQUE 9

Dynamic Programming: Computing a binomial coefficient -- Knapsack problem and memory functions -- Warshall's and Floyd's algorithm -- Greedy Technique: Prim's algorithm -- Kruskal's algorithm -- Dijkstra's algorithm.

UNIT IV INTERATIVE IMPROVEMENT AND BACKTRACKING 9

Iterative Improvement: The simplex method -- Maximum matching in bipartite graphs; Backtracking: N-queens problem – Hamiltonian circuit problem.

UNIT V LIMITATIONS OF ALGORITHM POWER 9

Branch and Bound: Knapsack problem -- Traveling salesman problem; Limitations of algorithm power: Lower-bound arguments -- P, NP and NP-complete problems; Coping with the Limitations of Algorithm Power: Approximation algorithms for NP-Hard problems -- Traveling salesman problem -- Knapsack problem.

TOTAL PERIODS(THEORY): 45

SUGGESTIVE EXPERIMENTS

1. Implementation of non-recursive and recursive algorithms for the given problem
2. Implementation of string matching using Brute Force technique
3. Implementation of Knapsack problem using Exhaustive Search technique
4. Implementation of merge sort and quick sort using Divide and Conquer technique
5. Implementation of Knapsack Problem using Dynamic Programming
6. Implementation of Prim's and Dijkstra's algorithms.
7. Implementation of n-Queens problem using Backtracking technique
8. Implementation of Knapsack using Branch and Bound technique
9. Mini project

TOTAL PERIODS(PRACTICAL): 30
TOTAL PERIODS: 75

OUTCOMES

On successful completion of this course, the student will be able to:

- Design algorithms for various computing problems (K3)
- Analyze the time and space complexity of algorithms (K4)
- Compare the different algorithm design techniques for a given problem (K4)
- Modify existing algorithms to improve efficiency (K4)
- Understand the limitations of algorithmic power (K2).

TEXTBOOKS

1. Anany Levitin, "Introduction to the Designand Analysis of Algorithms",3rd Edition,Pearson Education, 2012.
2. S Dasgupta, C H Papadimitriou, U V Vazirani, "Algorithms",1st Edition, McGraw-Hill Education, 2017.

REFERENCE BOOKS

1. ThomasH Cormen, Charles E Leiserson, RonaldL Rivest, CliffordStein, "Introduction to Algorithms", 3rd Edition,PHI Learning Private Limited,2012.
2. Steven S Skiena, "The Algorithm Design Manual", 2nd Edition,Springer,2008.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1404	DATABASE MANAGEMENT SYSTEMS	3	0	0	3

OBJECTIVES

- To introduce the fundamentals of database systems and conceptual modeling
- To learn SQL and relational model
- To learn database programming and relational database design
- To explore the fundamental concepts of transaction processing, concurrency control and recovery techniques
- To understand NOSQL databases.

UNIT I DATABASE SYSTEM CONCEPTS AND CONCEPTUAL MODELING 8

Characteristics of database -- Database users -- Data models, Schemas -- Three-schema architecture -- Database system environment -- Centralized and client/server DBMS architectures -- Data modeling using ER model -- Enhanced-ER model.

UNIT II RELATIONAL MODEL AND SQL 11

Relational Model: Concepts -- Constraints -- Update operations -- Constraint violations; Relational Algebra; Basic SQL: Data Definition -- SQL constraints -- Basic retrieval queries -- Insert, delete and update commands -- Schema modification; More SQL: dealing NULL -- Tuples, Set/Multiset comparisons -- Correlated nested -- EXISTS function -- SQL Renaming -- JOIN and Outer JOINs -- Aggregate functions -- Grouping.

UNIT III DATABASE PROGRAMMING AND DESIGN 10

Triggers -- Views; Database Programming Techniques: Overview and Issues -- Embedded SQL -- JDBC -- Database Stored Procedures and SQL/PSM; Design guidelines -- Functional dependencies -- First, second and third Normal Forms -- Boyce-Codd Normal Forms; FD: Inference rules -- Minimal cover; ER-to-relational mapping.

UNIT IV TRANSACTION, CONCURRENCY AND RECOVERY 8

Transaction Processing: Concepts -- ACID properties -- schedules -- serializability; Concurrency control: Two-phase locking technique -- Timestamp Ordering; Recovery: Concepts -- Deferred update -- Immediate update -- Multidatabase system recovery.

UNIT V NO SQL DATABASES 8

Introduction to NOSQL Systems -- The CAP Theorem -- Document-Based NOSQL Systems and MongoDB -- NOSQL Key -- Value Stores -- Column- Based or Wide Column NOSQL Systems -- NOSQL Graph Databases and Neo4j.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to:

- Understand the database concepts and apply ER models to any real-world application (K3)

- Apply data definition and complex SQL queries (K3)
- Design databases systematically and develop database software for various real-time applications (K3)
- Apply transaction processing, concurrency control and recovery mechanisms for practical problems (K3)
- Understand the concepts of NOSQL databases (K2)

TEXTBOOKS

1. RamezElmasri, Shamkant B Navathe, “Fundamentals of Database Systems”,7th Edition,Pearson, 2016.
2. Abraham Silberschatz, Henry F Korth, S Sudharshan, “Database System Concepts”, 6th Edition,Tata Mc Graw Hill, 2011.

REFERENCE BOOKS

1. Jeffrey D Ullman, Jennifer Widom, “A First Course in Database Systems”,3rd Edition,Pearson Education, 2014.
2. S Sumathi, S Esakkirajan, “Fundamentals of Relational Database Management Systems”, (Studiesin Computational Intelligence), Springer-Verlag,
1. 2007.
2. Raghu Ramakrishnan, “Database Management Systems”, 4thEdition, Tata Mc Graw Hill,2010.
3. C J Date, A Kannan, S Swamynathan, "An Introduction to Database Systems", 8th Edition, Pearson Education, 2006.
4. Hector Garcia-Molina, Jeffrey D Ullman, Jennifer Widom, "Database Systems: The Complete Book", 2e, Pearson.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1405	SOFTWARE ENGINEERING	3	0	0	3

OBJECTIVES

- To understand the phases in a software project
- To understand fundamental concepts of requirements engineering and Analysis Modeling.
- To understand the various software design methodologies
- To learn various testing techniques and maintenance measures.

UNIT I SOFTWARE PROCESS MODELS 9

Introduction to software engineering; Principles and practices; Software Process: Generic process model -- Perspective and specialized process models -- Secure development lifecycle; Introduction to Agility: Agile process model.

UNIT II PROJECT PLANNING AND MANAGEMENT 9

Project planning process; Software Project Estimation: Decomposition techniques -- Empirical estimation models -- The make/buy decision -- Project scheduling; Risk Management: Risk identification -- Risk projection -- Risk mitigation.

UNIT III REQUIREMENTS ANALYSIS AND SPECIFICATION 9

Software Requirements: Functional and non-functional -- User requirements -- System requirements -- Software requirements document; Requirement Engineering Process: Feasibility studies -- Requirements elicitation and analysis -- Requirements validation -- Requirements Management; Classical Analysis: Structured system analysis; Petri Nets.

UNIT IV SOFTWARE DESIGN 9

Design Concepts: Design process -- Design concepts -- Design model -- Modeling principles; Architectural Design: Architectural Styles --Architectural Mapping using Dataflow; User Interface Design: The Golden rules -- Interface Analysis -- Interface Design; Design for security; Component Level Design: Designing Class based Components -- Traditional Components.

UNIT V TESTING AND MAINTENANCE 9

Software Testing Fundamentals; Internal and External Views of Testing: White Box Testing -- Basis Path Testing -- Control Structure Testing-- Black Box Testing -- Unit Testing -- Integration Testing -- Regression Testing -- Validation Testing -- System Testing -- Security Testing; Debugging; Software Implementation: Coding Practices and Principles; Maintenance: Types -- Reengineering -- Reverse Engineering -- Restructuring.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to:

- Understand principles of software engineering and choose an appropriate process model (K4)

- Manage project schedule, estimate project cost and effort required (K3)
- Perform requirements analysis and modeling (K3)
- Apply systematic procedure for software design (K3)
- Compare and contrast the various testing and maintenance activities (K2).

TEXTBOOKS

1. Roger S Pressman, "Software Engineering – A Practitioner 's Approach", McGraw-Hill International Edition,SeventhEdition,2010.
2. Ian Sommerville, "Software Engineering", Pearson Education Asia, NinthEdition,2011.

REFERENCE BOOKS

1. Rajib Mall, "Fundamentals of SoftwareEngineering", PHI Learning Private Limited, Third Edition,2009.
2. PankajJalote, "Software Engineering, A Precise Approach", Wiley India,2010.
3. Kelkar S A,"Software Engineering", PrenticeHall of India,2007.
4. Stephen R Schach, "Software Engineering", Tata McGraw-Hill PublishingCompany Limited,2007.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1411	OPERATING SYSTEMS LAB	0	0	3	1.5

OBJECTIVES

- To learn and implement basic Unix commands usingsystem calls
- To implement various CPU Scheduling Algorithms
- To implement Process Creation and Inter Process Communication
- To implement Deadlock Avoidance and Deadlock Detection algorithms
- To implement Page Replacement Algorithms
- To implement File Organization and File Allocation Strategies

SUGGESTIVE EXPERIMENTS

1. Basics of UNIX commands and study of system calls.
2. Implement a few UNIX commands usingsystem calls.
3. Implement the various CPU Scheduling Algorithms
4. Implement Semaphores
5. Implement Sharedmemory and IPC
6. Implement Bankers Algorithm for Deadlock Avoidance
7. Implement Threading & Synchronization Applications
8. Implement the following Memory Allocation Methods for variable sized partition: a) First Fit b) Worst Fit c) Best Fit
9. Implement PagingTechnique of Memory Management
10. Implement the following Page Replacement Algorithms a) FIFO b) LRU c) LFU
11. Implement the various File Organization Techniques a) Single-levelb) Hierarchical
12. Implement thefollowing File Allocation Strategies a) Sequential b) Indexedc) Linked

TOTAL PERIODS: 45

OUTCOMES

After the completion of this course, students will be able to:

- Profile the performance of various CPU Scheduling Algorithms (K3)
- Implement Deadlock avoidance and Detection Algorithms (K3)
- Implement Semaphores and IPC (K3)
- Profile the performance of the various Page Replacement Algorithms (K3)
- Implement File Organization and File Allocation Strategies (K3)

LABORATORY REQUIREMENT FOR BATCH OF 38 STUDENTS

Hardware:

1. Standalone Desktops with Linux OS - 38 Nos

Software:

1. C/C++ Compiler

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1412	DATABASE LAB	0	0	3	1.5

OBJECTIVES

- To understand datadefinitions and datamanipulation commands
- To learn about the use of nested and join queries
- To understand procedural extensions of databases
- To design a database schema for an application using Normalization
- To implement typical database applications using Front-end tools

SUGGESTIVE EXPERIMENTS

1. Data Definition Commands
    - Creating tables with constraints, constraint violations
    - schema modifications
2. Data Manipulation Commands
    - Update operations
    - Simple SQL queries
    - Transaction Control statements - Savepoint and Rollback
3. Complex SQL Queries
    - Nested Queries
    - Correlated Subqueries
    - Joins and Outer Joins
    - Aggregate functions
    - Grouping and Ordering commands
4. Views
5. Database Programming: PL/SQL - Cursors
6. Procedures and Functions
7. Triggers
8. Exception Handling
9. Database design
    - ER Model, ER-to-relational mapping
    - Normalization
10. Implement the database application using any Front-end

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to:

- Use data definitions and manipulation commands(K3)
- Write complexSQL Queries(K3)
- Implement Database programming usingPL/SQL(K3)
- Design a database for real-time applications(K3)
- Implement and analyze the use of Tables, PL/SQL for a realistic database application using a Front-end tool(K3)

LABORATORY REQUIREMENT FOR BATCH OF 38 STUDENTS

Hardware:

1. Server - 1 Nos
2. Standalone Desktops - 38 Nos

Software:

1. Database: Oracle 10g - 38 Clientswith 1 Server Node
1. Frontend - NetBeans / Java / JDBC

COURSE CODE	COURSE TITLE	L	T	P	C
UEN1498	INTERPERSONAL SKILLS	0	0	2	1

OBJECTIVES

- To equip students with the adequate speaking and listening skills in English required for academic and general purposes.
- To impart speaking skills in English needed for both general and classroom conversation. This will include speaking for social needs (interaction with people in society for various purposes) as well as academic purposes (peer interaction, interaction with the teacher and the students in the class, facing interviews, interaction with teachers outside the class and others)
- To provide exposure to listening skills both general and academic. This will include listening in social situations and also in classroom contexts such as lectures, seminars and presentations,
- To focus on making academic presentations and also answering questionsin a job interview with confidence

UNIT I LISTENING FOR TAKING NOTES AND DEVELOPING IDEAS 6

Importance of listening and speaking skills for academic as well as general purposes -- giving and asking personal information -- expressing one's strengths and weaknesses -- ask for clarifications during a lecture -- listening to a lecture and taking notes -- listening to process descriptions and representing the same in appropriate flow charts -- Expand and express a complete idea avoiding fragmented utterances

UNIT II PRONUNCIATION IN CONTEXT 6

Conversation starters (small talk) -- pronunciation of individual sounds (vowels, consonants and diphthongs) -- syllable stress -- intonation patterns -- chunking for clarity -- compare and contrast information using relevant linking words -- converse on everyday topics with reasonable accuracy -- making presentations using PowerPoint slides.

UNIT III FLUENCY INCONTEXT 6

Fluency enhancement -- avoidance of factors affecting fluency -- practice in informal talks (greeting people -- responding to greetings -- describing places, objects, moods, health and symptoms -- inviting people -- expressing acceptance -- rejection -- other everyday functions)

UNIT IV ACTIVE LISTENING IN SPECIFIC SITUATIONS 6

What is active listening? Giving verbal and non-verbal feedback -- participation in group discussion -- Listening to Ted talks/talks by celebrities for the purposes of understanding, summarizing, appreciation, critical evaluation etc -- Practice in Role Plays.

UNIT V FORMAL AND INFORMALTALK 6

Vocabulary and tone in formal and informal situations- giving directions -- giving instructions in academic and business contexts -- advanced strategies for presentations and interactive communication -- Facing job interviews (FAQs and Mock Interview).

TOTAL PERIODS: 30

OUTCOMES

On successful completion of this course, the student will be able to:

- Listen and respond appropriately
- Participate in group discussions
- Make effective presentations
- Participate confidently and appropriately in conversations both formal and informal.
- Face campus interviews adequately.

TEXTBOOKS

1. Brooks, Margret, "Skills for Success, Listening andSpeaking, Level 4," Oxford University Press, Oxford,2011.
2. Richards C Jack, David Bholke, "Speak Now Level 3", Oxford University Press, Oxford,2010.

REFERENCE BOOKS

1. Bhatnagar, Nitin, Mamta Bhatnagar, "Communicative English for Engineers and Professionals", Pearson, New Delhi, 2010.
2. Hughes, Glyn, Josephine Moate, "Practical English Classroom", Oxford University Press, Oxford,2014.
3. Vargo, Mari, "Speak Now Level 4", Oxford University Press, Oxford, 2013.
4. Richards C Jack, "Person to Person(Starter)", OxfordUniversity Press, Oxford,2006.
5. Ladousse, Gillian Porter, "Role Play", Oxford University Press,Oxford,2014.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1501	COMPUTER NETWORKS	3	0	0	3

OBJECTIVES

- To understand the protocol layering and physical level communication
- To understand the various components required to build different networks
- To learn the functions of network layer and the various routing protocols
- To familiarize the functions and protocols of the Transport layer
- To understand various application layer protocols.

UNIT I INTRODUCTION AND PHYSICAL LAYER 9

Introduction: Networks -- Network types -- Protocol layering -- TCP/IP Protocol suite -- OSI model; Physical Layer: Performance; Socket Programming; Transmission media -- Switching -- Circuit-switched networks -- Packet switching.

UNIT II DATA-LINK LAYER AND MEDIA ACCESS 9

Introduction: Link-Layer addressing -- DLC services -- Data-Link layer Protocols -- HDLC -- PPP -- Media access control -- Wired LANs: Ethernet -- Wireless LANs -- Introduction -- IEEE 802.11 – Bluetooth -- Connecting devices.

UNIT III NETWORK LAYER 9

Network layer services -- Packet switching -- Performance -- IPV4 Addresses -- Forwarding of IP packets; Network Layer Protocols: IP -- ICMP v4 -- Unicast routing algorithms -- Protocols – Multicasting basics -- IPV6 addressing -- IPV6 protocol.

UNIT IV TRANSPORT LAYER 9

Introduction: Transport layer protocols -- Services -- Port numbers -- User datagram protocol -- Transmission control protocol -- SCTP.

UNIT V APPLICATION LAYER 9

Traditional applications -- Electronic mail (SMTP, POP3, IMAP, MIME) -- HTTP -- File transfer protocol -- Secure shell (SSH) -- DNS -- SNMP.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to:

- Understand the principles, design, terminology and concepts of the network models (K2)
- Learn the protocols of data link layer and apply error control mechanisms (K3)
- Understand the core functions of network layer protocols and apply them for data communication (K3)
- Learn the purpose of Transport Layer Protocols such as UDP and TCP (K2)
- Understand the Application Layer Protocols and their basic functionalities (K2)

TEXTBOOKS

1. Behrouz A Forouzan, “Data Communications and Networking”, 5th Edition TMH, 2013.
2. LarryL Peterson, Bruce S Davie, “Computer Networks: A Systems Approach”, 5th Edition,Morgan Kaufmann Publishers, 2012.

REFERENCE BOOKS

1. WilliamStallings, “Data and Computer Communications”, 10th Edition, Pearson Education, 2013.
2. Nader F Mir, “Computer and Communication Networks”, 2nd Edition, Prentice Hall, 2014.
3. Ying-DarLin, Ren-Hung Hwang, Fred Baker, “Computer Networks: An Open Source Approach”, McGraw-Hill Publisher, 2011.
4. James F Kurose, Keith W Ross, “Computer Networking, A Top-Down Approach Featuring the Internet”, 6th Edition,Pearson Education, 2013.
5. Andrew Tanenbaum, David J Wetherall, “Computer Networks”, 5th Edition, Pearson Education, 2013.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1502	MICROPROCESSORS AND INTERFACING	3	0	0	3

OBJECTIVES

- To understand the Architecture of 8086 microprocessor
- To learn the design aspects of I/O and Memory Interfacing circuits
- To interface microprocessors with supporting chips
- To study the Architecture of 8051 microcontroller
- To design a microcontroller-based system.

UNIT I THE 8086 MICROPROCESSORS 9

Introduction to 8086 -- Microprocessor architecture -- Addressing modes -- Instruction set and assembler directives -- Assembly language programming -- Stacks -- Procedures -- Macros -- Interrupts and interrupt service routines -- Byte and String Manipulation.

UNIT II 8086 SYSTEM BUS STRUCTURE 9

8086 signals -- Basic configurations -- System bus timing -- System design using 8086 -- I/O programming -- Introduction to Multiprogramming -- System Bus Structure -- Multiprocessor configurations -- Coprocessor, closely coupled and loosely Coupled configurations -- Introduction to advanced processors.

UNIT III I/O INTERFACING 9

Memory interfacing and I/O interfacing -- Parallel communication interface -- Serial communication interface -- D/A and A/D Interface -- Timer -- Keyboard /display controller -- Interrupt controller -- DMA controller -- Programming and applications Case studies: Traffic Light control, LED display, LCD display, Keyboard display interface and Alarm Controller.

UNIT IV THE 8051 MICROCONTROLLERS 9

Architecture of 8051 -- Special Function Registers(SFRs) -- I/O Pins ports and circuits -- Instruction set -- Addressing modes -- Assembly language programming.

UNIT V INTERFACING MICROCONTROLLER 9

Programming 8051 Timers -- Serial port programming -- Interrupts programming -- LCD & keyboard interfacing -- ADC, DAC & Sensor interfacing -- External memory interface -- Stepper motor and waveform generation -- Comparison of microprocessor, microcontroller, PIC and ARM processors.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to:

- Understand the basic architecture, operation, programming of microprocessor 8086 (K3)
- Understand the design of basic and multiprocessor systems and their bus timings (K2)
- Design the 8086 interfaces with memory, I/O and other peripheral chips (K3)
- Understand the basic architecture and operation of microcontroller 8051 (K2)
- Apply programming concepts to implement microcontroller interfaces for different applications (K3).

TEXTBOOKS

1. Doughlas V Hall, “Microprocessors and Interfacing, Programming and Hardware”, TMH, 2012.
2. Mohamed Ali Mazidi, Janice Gillispie Mazidi, Rolin McKinlay, “The 8051Microcontroller and Embedded Systems: Using Assembly and C”, 2nd Edition, Pearson education, 2011.

REFERENCE BOOKS

1. Yu-Cheng Liu, Glenn A Gibson, “Microcomputer Systems: The 8086/8088Family-Architecture, Programming andDesign”, 2ndEdition,Prentice Hall of India,2007.
2. A K Ray, K M Bhurchandi, “Advanced Microprocessors and Peripherals”,3rd edition, Tata McGraw-Hill, 2012.
3. Barry B Bray, “The Intel Microprocessor 8086/8088,80186,80286,80386 and80486 - Architecture, Programming and Interfacing”, 8th Edition, PHI, 2011.
4. Mohamed Rafiquazzaman, “Microprocessor and Microcomputer based System Design”, 2nd Edition,Universal Book Stall, 1995.
5. Kenneth J Ayala,“The 8051 Microcontroller Architecture, Programming and Applications”, 2nd edition, Penram International, 1996.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1203	THEORY OF COMPUTATION	3	2	0	4

OBJECTIVES

- To construct finite automata for any given pattern and find its equivalent regular expressions
- To understand the language hierarchy and to design a context free grammar for any given language
- To construct pushdown automata for any CFL
- To understand Turing machines and their capability
- To understand undecidable problems.

UNIT I FINITE AUTOMA AND REGULAR LANGUAGES 10

Basic Mathematical Notation and Techniques; Finite Automata (FA): Deterministic Finite Automata (DFA) -- Non-deterministic FiniteAutomata (NFA) -- Finite automata with epsilon transitions -- Equivalence of FAs -- Minimization of DFA; Regular Expressions andLanguages: Regular expressions -- Finite automata and regular expressions; Properties of Regular Languages: Proving languages not to be regular -- Closure and decision properties of regular languages.

UNIT II CONTEXT FREE GRAMMARS AND LANGUAGES 8

Chomsky's Hierarchy of Languages; Context-Free Grammar and Languages: Context-Free Grammar (CFG) -- Parse trees -- Ambiguity in grammars and languages; Normal Forms for Context Free Grammars: Eliminating useless symbols -- Computing the generating and reachable symbols -- Eliminating null productions -- Eliminating unit productions -- Chomsky Normal Form (CNF) -- Greibach Normal Form (GNF).

UNIT III PUSHDOWN AUTOMATA 9

Pushdown Automata (PDA): Definition of the Pushdown automaton -- The languages of a PDA -- Equivalence of PDAs and CFGs -- Deterministic Pushdown automata; Pumping Lemma for Context Free Languages.

UNIT IV TURNING MACHINES 9

The Turing Machine -- Programming Techniques for Turing Machines -- Extensions to the Basic Turing Machine -- Restricted Turing Machines.

UNIT V UNDECIDABILITY 9

Undecidability: Language that is not Recursively Enumerable (RE) -- Undecidable problem that is RE -- Undecidable problems about Turing machines -- Post's Correspondence Problem (PCP) -- Other undecidable problems.

TOTAL PERIODS: 60

OUTCOMES

On successful completion of this course, the student will be able to:

- Construct automata, regular expression for any given pattern (K3)

- Understand the need of formal languages, and grammars (K3)
- Design pushdown automata for any CFL (K3)
- Design Turing machines for any Languages (K3)
- Explain the Decidability or Undecidability of various problems (K2).

TEXTBOOKS

1. Hopcroft J E, Motwani R, UllmanJ D, "Introduction to Automata Theory, Languages and Computations", Pearson Education, 3rd Edition,2008.

REFERENCE BOOKS

1. Harry R Lewis,Christos H Papadimitriou, "Elements of theTheoryofComputation", PrenticeHall of India,2nd Edition,2003.
2. Peter Linz, "An Introduction to Formal Language and Automata", NarosaPublishers, 3rd Edition,2002.
3. MishraK L P, Chandrasekaran N, "Theoryof Computer Science – Automata, Languages and Computation", Prentice Hall of India, 3rd Edition,2004.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1504	ARTIFICIAL INTELLIGENCE	3	0	2	4

OBJECTIVES

- To Study the fundamental concepts of AI agents and environments.
- To Learn the methods of problem solving in AI using various search strategies.
- To Understand the concepts of knowledge representation and inference using logic.
- To Understand the concepts of knowledge representation and inference under uncertainty.
- To Learn the introductory concepts of machine learning in AI.

UNIT I FOUNDATIONS 8

Introduction: What is AI; Intelligent Agents: Agents and environments-- Good behavior -- The nature of environments -- Structure of agents;Philosophical Foundations: Weak AI -- Strong AI -- Ethics and risks ofdeveloping AI; AI: The Present and Future: Agent components -- Agent architectures.

UNIT II PROBLEM SOLVING & SEARCH TECHNIQUES 10

Solving Problems by Searching: Problem solving agents – Example problems -- Searching for solutions -- Uninformed search strategies -- Informed search strategies -- Heuristic functions; Beyond classical search: Local search algorithms and optimization problems; Adversarial search: Games -- Optimal decisions in games -- Alpha-beta pruning.

UNIT III KNOWLEDGE REPRESENTATION & REASONING 9

Logical Agents: Knowledge-based agents -- Propositional logic -- Propositional theorem proving; First order logic: Syntax and semantics for first order logic -- Using first order logic; Inference in first order logic: Propositional versus first order logic -- Unification and lifting -- Forward chaining -- Backward chaining -- Resolution.

UNIT IV UNCERTAIN KNOWLEDGE AND REASONING 9

Quantifying Uncertainty: Acting under uncertainty -- Basic probability notation -- Inference using full joint distributions -- Bayesian rule & its use; Probabilistic Reasoning: The semantics of Bayesian networks -- Exact inference in Bayesian networks -- Other approaches to uncertain reasoning.

UNIT V LEARNING 9

Learning from Examples: Forms of learning -- Supervised learning -- Learning decision trees; Reinforcement learning: Passive reinforcement learning -- Active reinforcement learning -- Application to robot control.

TOTAL PERIODS(THEORY): 45

SUGGESTED LAB EXERCISES

1. Uninformed Search Techniques
2. Informed Search Techniques
3. Hill Climbing algorithms
4. Adversarial Search techniques

5. Construction of AND-OR graph from knowledge base
6. Inference from knowledge base
7. Inference usingfull joint probability distribution
8. Inference usingBayesiannetwork
9. Decisiontree learning algorithm
10. Passivereinforcement learning algorithm

TOTAL PERIODS(PRACTICAL): 30
TOTAL PERIODS: 75

OUTCOMES

On successful completion of this course, the student will be able to

- Identify, formulate, understand and solve AI problems using search techniques (K3)
- Elucidate the concept of Knowledge Representation and inference using logics (K2)
- Elucidate the concept of Knowledge Representation and inference under uncertainty (K2)
- Elucidate the concept of learning in AI applications (K3)
- Implement various search, inference and learning algorithms in AI (K4)

TEXTBOOKS

1. Stuart Russell, Peter Norvig, "Artificial Intelligence – A Modern Approach", 3rdEdition, Pearson Education / PrenticeHall of India,2015.
2. Deepak Khemani "A First Course in Artificial Intelligence", McGraw Hill,2014.

REFERENCE BOOKS

1. DawnW Patterson, "Introduction toArtificialIntelligence andExpert Systems", 1st Edition,Pearson Education India,2015.
2. Elaine Rich, Kevin Knight, "Artificial Intelligence", 2nd Edition, Tata McGraw- Hill, 2003.
3. Andreas Muller,SarahGuido, "Introduction to Machine Learning with Python: A Guidefor Data Scientists", Shroff/O'Reilly, 1st edition, 2016.
4. DavidPoole,AlanMackworth, "ArtificialIntelligence:Foundationof Computational Agents", 2nd Edition,Cambridge University Press, 2017.
5. Prateek Joshi, "Artificial Intelligence with Python", 1st edition, Packt PublishingLimited, 2017.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1505	INTRODUCTION TO CRYPTOGRAPHIC TECHNIQUES	3	0	0	3

OBJECTIVES

- To understand the classical and symmetric cryptographic techniques
- To study about message authentication and hash functions
- To learn number theory fundamentals needed by cryptographic algorithms
- To understand the various key distribution and management schemes
- To understand the concepts of Public key cryptography and digital signatures.

UNIT I INTRODUCTION AND CLASSICAL CRYPTOGRAPHY AND SYMMENTRIC CRYPTOGRAPHY 10

Cryptography and Modern Cryptography -- Setting of Private-Key Encryption -- Historical Ciphers -- Basic Principles; Perfectly Secret Encryption; Private-Key Encryption and Pseudo randomness.

UNIT II MESSAGE AUTHENTICATION CODES AND COLLISION-RESISTANT HASH FUNCTIONS 8

Secure Communication and Message Integrity -- Encryption vs. Message Authentication -- Message Authentication Codes -- Constructing Secure Message Authentication Codes -- CBC-MAC -- Collision-Resistant Hash Functions -- NMAC and HMAC -- Constructing CCA-Secure Encryption Schemes -- Obtaining Privacy and Message Authentication.

UNIT III BLOCK CIPHERS 10

Substitution-Permutation Networks -- Feistel Networks -- DES -- AES -- Differential and Linear Cryptanalysis; One-Way Functions -- From One-Way Functions to Pseudo randomness -- Constructing Pseudorandom Generators -- Constructing Pseudorandom Permutations -- Necessary Assumptions for Private-Key Cryptography.

UNIT IV NUMBER THEORY & KEY DISTRIBUTION 8

Number Theory: Preliminaries and Basic Group Theory -- Primes, Factoring, and RSA -- Cryptographic Applications of Number-Theoretic Assumptions; Private-Key Management and the Public-Key Revolution: Limitations of Private-Key Cryptography -- Key Distribution Centers -- The Public-Key Revolution -- Diffie-Hellman Key Exchange.

UNIT V PUBLIC-KEY ENCRYPTION & DIGITAL SIGNATURE 9

Public-Key Encryption – An Overview -- Definitions -- Hybrid encryption -- RSA encryption – The El Gamal Encryption Scheme -- Security Against Chosen-Ciphertext Attacks; Digital Signatures Schemes: An Overview -- Definitions -- RSA Signatures -- The Hash-and-Sign Paradigm -- Lamport's One-Time Signature Scheme -- Signatures from Collision -- Resistant Hashing -- The Digital Signature Standard -- Certificates and Public-Key Infrastructures; Authentication Protocol: SSL and TLS.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to:

- Describe and implement classical and symmetric ciphers (K2)
- Describe the authentication schemes and hash algorithms (K2)
- Understand the number theoretic foundations of cryptography (K3)
- Compare and contrast various Public key cryptographic techniques (K3)
- Illustrate various Public key cryptographic techniques (K3).

TEXTBOOKS

1. Jonathan Katz, YehudaLindell, "Introduction to Modern Cryptography",2nd Edition (Chapman & Hall/CRC Cryptography and Network Security Series), 2014.
2. Wenbo Mao, "Modern Cryptography – Theory and Practice", Pearson Education, 2004.

REFERENCE BOOKS

1. Johannes A Buchmann, "Introduction to Cryptography", 2nd edition, Pearson Education, Springer, 2009.
2. Charles P Pfleeger, Shari Lawrence Pfleeger, "Security in computing", 3rdEdition,PrenticeHall of India,2006.
3. Bruce Schneier,Neils Ferguson, "Practical Cryptography",1st Edition, Wiley Dreamtech India Pvt Ltd, 2003.
4. http://nptel.ac.in/courses/106105031/lecture by Dr Debdeep Mukhopadhyay, IIT Kharagpur.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1511	NETWORKS LAB	0	0	3	1.5

OBJECTIVES

- To learn and use network commands.
- To learn socket programming.
- To implement and analyze various network protocols.
- To learn and use simulation tools.
- To use simulation toolsto analyze theperformance of various network protocols.

SUGGESTIVE LAB EXERCISES

1. Learn to use commands like tcpdump, netstat, ifconfig, nslookup and traceroute. Capture ping and traceroute PDUs using a network protocol analyzer and examine.
2. Write a HTTP web client program to download a web page using TCP
3. sockets.
4. Write applications using TCP sockets like:
    a. Echo client and echo server
    b. Chat
    c. File Transfer
5. Simulate of DNS using UDP sockets.
6. Write programs simulating ARP /RARP protocols.
7. Study of Network simulator (NS) and Simulation of Congestion
8. Control Algorithms using NS.
9. Study of TCP/UDP performance using Simulation tool.
10. Simulate Distance Vector/ Link State Routing algorithm.
11. Evaluate the performance of Routing protocols using Simulation tool.
12. Simulate of error correction code (like CRC)

TOTAL PERIODS: 60

OUTCOMES

On successful completion of this course, the student will be able to:

- Implement various protocols using TCP and UDP (K2)
- Compare the performance of different transport layer protocols (K2)
- Use simulation tools to analyze the performance of various network protocols (K3)
- Analyze various routing algorithms (K3)
- Implement error correction codes (K2)

LABORATORY REQUIREMENT FOR BATCH OF 38 STUDENTS

Hardware:

1. Standalone Desktops - 38 Nos

Software:

1. C / C++ / Java / Python / Equivalent Compiler
2. Network simulator like NS2 / NS3 / GlomoSim / OPNET/ Packet Tracer/ Equivalent

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1512	MICROPROCESSORS LAB	0	0	3	1.5

OBJECTIVES

- To understand simple Assembly Language Programs concepts and features
- To write Assembly Language Programming for 8086
- To understand MASM programming
- To design different, I/O interfaces with Microprocessors
- To write Assembly Language Programming for 8051

SUGGESTIVE LIST OF EXPERIMENTS

8086 Programs usingkits

1. Basic arithmetic and Logical operations
2. Move a datablock without overlap
3. String manipulations and sorting and searching
4. Code conversion, decimal arithmetic and Matrix operations.

8086 Programs usingMASM

1. Floatingpointoperations
2. Password checking, Print system date and time
3. Counters and Time Delay

Peripherals and Interfacing Experiments

1. Traffic light controller
2. Stepper motorcontrol
3. Keyboard and Display controller
4. Serial interface and Parallelinterface
5. A/D and D/A interface and Waveform Generation

8051 Experiments usingkits

1. Basic arithmetic and logical operations
2. Squareand Cube program, 2‘s complement of a number
3. Unpacked BCD to ASCII

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to:

- Write 8086 Assembly Language Programs using arithmetic, logical and string operations (K3)
- Implement BIOS interrupt programming using MASM (K3)
- Design different basic I/O interfaces with 8086 (K3)
- Design different real-time interfaces with 8086 (K3)
- Write 8051 Assembly Language Programs using arithmetic and logical operations (K3)

LABORATORY REQUIREMENT FOR BATCH OF 38 STUDENTS

Hardware:

1. Standalone Desktops - 30 Nos
2. 8086 development kits - 30 Nos
3. Interfacing Units - Each 10 Nos
4. Microcontroller kits- 30 Nos

Software:

1. MASM software

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1601	INTERNET PROGRAMMING	3	0	0	3

OBJECTIVES

- To understand different client and serverend technologies
- To develop web application usingtraditional technologies
- To understand anddevelop web applications usingemerging web application frameworks.

UNIT I WEB ESSENTIALS 9

Web Essentials: Clients -- Servers -- Communication; Basic internet protocols -- World wide web -- HTTP Request Message -- HTTP Response Message -- Web Clients -- Web Servers; Web Server Security: Web server vulnerabilities -- Effects and types of attacks -- Avoiding attacks on web servers; HTML5: Tables -- Lists -- Image -- Redefined form elements -- Semantic elements -- Audio and video controls -- CSS3: Inline -- Embedded -- External style sheets -- Rule cascading -- Inheritance -- Adding graphics to web pages -- Transformations -- Transitions -- Animations.

UNIT II CLIENT-SIDE PROGRAMMING 9

Client-side programming language -- JavaScript: Variables – Data types -- Statements -- Operators -- Objects -- Arrays -- Built-in objects -- Event handling -- DOM: Introduction -- Document tree -- Node object -- Document -- Element -- Text nodes -- Event handling -- Event propagation

UNIT III SERVER-SIDE PROGRAMMING 9

Servlets: Architecture -- Life Cycle -- Parameter data -- Sessions -- Cookies and URL rewriting -- AJAX: Ajax Client Server Architecture -- XML Http Request Object -- Call Back Methods.

UNIT IV CLIENT-SIDE FRAMEWORK 9

Introduction to ReactJS –- Pure react: React elements – ReactDOM –-Children –– Constructing elements with data –– React components;Props, state and component tree –- React router: Incorporation –-Nesting –– Parameters; Communication of react with server.

UNIT V SERVER-SIDE FRAMEWORK 9

Introduction to NodeJS –- Event Loop –– Patterns –– Core APIs: Event Emitter -- Callback syntax -- Streams -- File system, -- Buffers -- Process module -- Data Access –– NoSQL and Document stores: MongoDB

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to:

- Design modern websites using HTML5 and CSS3(K3)
- Design and implement dynamic web pages with JavaScript and DOM (K3)
- Develop responsive web applications using Servlets and AJAX (K3)
- Develop web applications using ReactJS framework (K3)
- Develop web applications using NodeJS framework (K3)

TEXTBOOKS

1. Jeffrey C, Jackson, "Web Technologies A Computer Science Perspective", Pearson Education, 2011 (Unit 1,2,3)
2. Alex Banks, Eve Porcello, "Learning React: Functional Web Development with React and Redux", O'Reilly Media Inc., May 2017 (Unit 4)

REFERENCE BOOKS

1. Tom Hughes-Crouchers and Mike Wilson, "Node: Up and Running", 1st Edition, O'Reilly Media,2012 (Unit 5)
2. Uttam K Roy, "Web Technologies", Oxford University Press, 2010. (Unit 3)
3. Matthew MacDonald, "HTML5:The missing manual", O'Reilly Media, August 2011(Unit 1)
4. "How to Hack a Web Server" – https://www.guru99.com/how-to-hack-web-server.html (Unit 1)
5. David McFarland, "CSS3: The missing manual", O'Reilly Media, December 2012 (Unit 1)

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1602	COMPILER DESIGN	3	0	2	4

OBJECTIVES

- To learn various phases of a compiler
- To learn various parsing techniques
- To understand intermediate code generation
- To learn to implement code generator and optimize the code.

UNIT I INTRODUCTION TO COMPILERS 9

Language processors -- Phases of compiler -- Role of lexical analyzer -- Input buffering -- Specification of tokens -- Recognition of tokens; Lexical analyzer generator: Structure of lex program – Look ahead operator and conflict resolution.

UNIT II SYNTAX ANALYSIS 9

Role of Parser -- Writing grammars for language constructs -- Types of grammars: Ambiguity -- Deterministic and recursive; Top down parsers: Recursive descent parser -- Predictive parser; Bottom up parsers: SLR Parser -- CLR Parser -- LALR Parser; Error handling and recovery in syntax analyzer; Syntax analyzer generator: Structure of yacc program -- Creating =yacc= lexical analyzers with =lex=.

UNIT III INTERMEDIATE CODE GENERATION 9

Syntax directed definitions: Synthesized attribute -- Inherited attribute -- Dependency graph -- Evaluation order of syntax directed definitions; Intermediate languages: Syntax tree -- Three address code; SDD for type checking -- Declarations -- Evaluation of expressions and flow of control statements -- Bottom-up evaluation of S-attribute definitions.

UNIT IV RUNTIME ENVIRONMENTS AND CODE GENERATION 9

Source language issues -- Storage organization -- Storage allocation strategies: Static, Stack and Heap -- Implementation of symbol table -- Issues in code generation -- Design of a simple code generator.

UNIT V CODE OPTIMIZATION 9

Principal sources of optimization -- DAG -- Optimization of basic blocks -- Global data flow analysis -- Introduction to Low Level Virtual Machine (LLVM) -- Design of LLVM -- Core libraries -- Developing plugin in LLVM.

TOTAL PERIODS(THEORY): 45

SUGGESTIVE EXPERIMENTS

1. Implementation of Lexical Analyzer using Lex Tool
2. Implementation of Arithmetic Calculator using LEX and YACC
3. Generation of TAC for a simple program using LEX and YACC
4. Consider a simple program as an input and process this code to print the intermediate code after every phase. It is necessary to print the output of lexical, syntax, semantic, intermediate code generation, code optimization and code generation phases
5. Study of LLVM framework.

TOTAL PERIODS(PRACTICAL): 30
TOTAL PERIODS: 75

OUTCOMES

On successful completion of this course, the student will be able to:

- Design a lexical analyzer for a sample language (K3)
- Apply different parsing algorithms to develop the parsers for the given grammar (K3)
- Write syntax directed translation for programming language constructs (K3)
- Understand and implement a simple code generator (K3)
- Understand and implement code optimization techniques (K3).

TEXTBOOKS

1. Alfred V Aho, Monica S Lam, Ravi Sethi, Jeffrey D. Ullman,"Compilers: Principles, Techniques and Tools", 2nd Edition,Pearson Education, 2009.

REFERENCE BOOKS

1. Randy Allen, Ken Kennedy, "Optimizing Compilers for Modern Architectures:A Dependence BasedApproach", Morgan Kaufmann Publishers,2002.
2. Steven S. Muchnick, "Advanced Compiler Design and Implementation", Morgan Kaufmann Publishers – Elsevier Science, India, Indian Reprint, 2003.
3. Keith D Cooper, Linda Torczon, "Engineering a Compiler", Morgan Kaufmann Publishers Elsevier Science, 2004.
4. Andrew W. Appel,"Modern Compiler Implementation in C", CambridgeUniversity Press, 1st edition, 2004.
5. Watson,Des, "A PracticalApproach to Compiler Construction", 1st edition, Springer-Verlog, 2017.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1603	INTRODUCTION TO MACHINE LEARNING	3	0	0	3

OBJECTIVES

- To understand machine learning problems
- To study the various supervised, unsupervised and reinforcement learning algorithms in machine learning
- To study the dimensionality reduction techniques to represent the data and their dependencies
- To understand the need of optimization techniques.

UNIT I INTRODUCTION 8

Introduction: Machine learning; Examples of Machine Learning Applications: Learning associations -- Classification -- Regression -- Unsupervised learning -- Reinforcement learning; Preliminaries: Weight space -- Curse of dimensionality -- Testing machine learning algorithms -- Turning data into probabilities -- Basic statistics -- Bias-variance tradeoff.

UNIT II SUPERVISED LEARNING 11

Neural Networks and Linear Discriminants: Brain and the neuron -- Neural networks -- Perceptron -- Linear separability – Linear regression; Multi-layer Perceptron: Going forward -- Back-propagation of error; Support Vector Machines.

UNIT III PROBABILISTIC LEARNING, LEARNING WITH TREES 9

Probabilistic Learning: Gaussian mixture models -- Nearest neighbour methods; Learning with Trees: Constructing decision trees -- Classification and Regression trees -- Classification example; Ensemble Learning: Boosting -- Bagging -- Random forests.

UNIT IV UNSUPERVISED LEARNING, REINFORCEMENT LEARNING 9

Unsupervised: K-means algorithm -- Self-organizing feature map; Reinforcement learning: State and action space -- Reward function -- Discounting -- Action selection -- Policy -- Markov decision process -- Values -- SARSA and Q-learning.

UNIT V DIMENSIONALITY REDUCTION, OPTIMISATION TECHNIQUES 8

Dimensionality Reduction Techniques: Linear Discriminant analysis, Principal Component Analysis; Optimization and Search: Least-squares optimization -- Conjugate gradients -- Search approaches -- Exploitation and exploration.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to:

- Understand the basic concepts of machine learning (K2)
- Apply supervised algorithms for different classification problems (K3)
- Understand the need of ensemble methods (K2)
- Apply unsupervised and reinforcement learning techniques to various problems (K3)

- Understand the requirement of dimensionality reduction and optimization techniques (K2)

TEXTBOOKS

1. Stephen Marsl and, “Machine Learning – An Algorithmic Perspective”, 2nd Edition, Chapman and Hall/CRC Machine Learning and Pattern RecognitionSeries, 2015.
2. Ethem Alpaydin, “Introduction to Machine Learning”, 3rdEdition, The MIT Press, 2014.

REFERENCE BOOKS

1. Jason Bell, “Machine learning – Hands on for Developers andTechnicalProfessionals”, 1st Edition,Wiley, 2014.
2. Peter Flach, “Machine Learning: The Art andScience of Algorithms thatMake Sense of Data”,1st Edition,Cambridge University Press, 2012.
3. Richert,Willi, “Building machine learning systems withPython”, PacktPublishing, 2013.
4. TomM Mitchell,“Machine Learning”, McGraw-Hill Education (India),2013.
5. Y S Abu-Mostafa, M Magdon-Ismail, HT Lin, “Learning from Data”, AML Book Publishers, 2012.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1604	OBJECT ORIENTED ANALYSIS AND DESIGN	3	0	0	3

OBJECTIVES

- To understand and differentiate Unified Process from other approaches
- To understand object-oriented software design using UML's static diagrams
- To understand software modeling using the UML's dynamic diagrams
- To learn improving software design with design patterns
- To learn testing the software with its requirements specification.

UNIT I DEVELOPMENT PROCESS & USE-CASE DIAGRAM 9

Introduction to OOAD with OO Basics -- Unified process -- UML diagrams -- Use case -- Case study -- The Next Gen POS system, Inception – Use case modelling -- Relating use cases -- Include, extend and generalization -- When to use use-cases.

UNIT II UNIFIED PROCESS & CLASS DIAGRAM 9

Class diagram -- Elaboration -- Domain model -- Finding conceptual classes and description classes -- Associations -- Attributes -- Domain model refinement -- Finding conceptual class hierarchies -- Aggregation and composition -- Relationship between sequence diagrams and use cases -- When to use class diagrams.

UNIT III DYNAMIC & IMPLEMENTATION DIAGRAMS 9

Dynamic Diagrams: UML interaction diagrams -- System sequence diagram -- Collaboration diagram -- When to use communication diagrams -- State machine diagram and modelling -- When to use state diagrams -- Activity diagram -- When to use activity diagrams. Implementation Diagrams: UML package diagram -- When to use package diagrams -- Component and deployment diagrams -- When to use component and deployment diagrams.

UNIT IV DESIGN PATTERNS 9

Designing objects with responsibilities -- Creator – Information expert -- Low coupling -- High cohesion -- Controller design patterns -- Creational -- Factory method -- Structural -- Bridge -- Adapter -- Behavioural -- Strategy -- Observer -- Applying GoF design patterns -- Mapping design to code.

UNIT V TEST DRIVEN DEVELOPMENT AND REFACTORING 9

Object oriented methodologies -- Software quality assurance – Impact of object orientation on testing -- Develop test cases and test plans.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Express software design with UML diagrams (K2)
- Design and implement projects using OO concepts (K4)
- Identify and map basic software requirements in UML mapping (K3)

- Transform UML based software design into pattern based design using design patterns (K4)
- Test any object-oriented software against its requirements (K3).

TEXTBOOKS

1. Larman, Craig, "Applying UML and Patterns", Pearson Education Asia,2008.
2. Ali Bahrami, "Object Oriented Systems Development", McGraw Hill International Edition,1999.

REFERENCE BOOKS

1. Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, "Design patterns: Elements of Reusable Object-Oriented Software", Addison Wesley, 1995.
2. Martin Fowler, "UML Distilled: A Brief Guide to the Standard Object Modeling Language", 3rd edition, Addison Wesley, 2003.
3. Booch, G, Jacobson I, Rumbaugh J, "The Unified Modeling Language UserGuide", Addison Wesley, 2008.
4. Roger S Pressman, "Software Engineering – A Practitioner 's Approach",7th edition, 2010.
5. Aditya P Mathur, "Foundations of SoftwareTesting– FundamentalAlgorithms andTechniques", Dorling Kindersley (India)Pvt.Ltd., Pearson Education, 2008.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1611	INTERNET PROGRAMMING LAB	0	0	3	1.5

OBJECTIVES

- To develop web applications using traditional client and server end technologies.
- To develop web applications using JavaScript frameworks.

SUGGESTIVE LAB EXPERIMENTS

- Createa webpageusingTable,Lists, Imageandanchorelements.(Ex: Biodata)
- Createwebsite for the International Conference usingHTML5 andCSS3 elements.
- Validate a registration form using JavaScript event handling mechanisms.
- Develop a web application to authenticate the user with servlet and MySQL.
- Develop a web application usingsessionsandservlet.(Ex: Online Shopping application)
- Develop a web application usingservlet,cookies and MySQL. (Ex: SuperMarketBilling System)
- Develop a Popup MenuApplication usingAJAX.
- Develop a web application using ReactJS and NodeJS. (Ex: Railway Reservation System)

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Designwebsites usingHTML5 and CSS3 (K3)
- Develop multi-tier web application usingServlets (K3)
- Develop interactive web applications usingAJAX (K3)
- Develop web applications usingAngularJSand NodeJS frameworks (K3)

LABORATORY REQUIREMENT FOR A BATCH OF 38 STUDENTS

Hardware:

1. Standalone Desktops - 38 Nos

Software:

1. Frontend - NetBeans/Eclipse with Java
2. Database - MySQL
3. Web Server - Tomcat Server

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1617	MINI PROJECT	0	0	3	1.5

OBJECTIVES

- To capture the requirements specification for an intended software system
- To draw the UML diagrams for the given specification
- To map the design properly to code
- To test the software system thoroughly for all scenarios
- To improve the design by applying appropriate design patterns.

SUGGESTIVE LIST OF EXPERIMENTS

Draw standard UML diagrams using an UML modeling tool for a given case study and map design to code and implement a 3 layered architecture. Test the developed code and identify whether the SRS is satisfied.This could be done in an incremental way and the next increment should undergo the same process of design, coding and testing to satisfy the SRS. Any one of the suggested domains could be chosen for the mini project.The student could also choose his own interested domain base on the approval from the concerned faculty.

1. Develop a problem statement.
2. Document theSoftwareRequirements Specification (SRS) for theabove problem statement.
3. Identify use cases and develop the Use Case model.
4. Identify theconceptual classesanddevelop a Domain Modeland derivea Class Diagram from the already designed Domain Model.
5. Usingtheidentified scenarios, findtheinteraction between objectsand represent themusingUML Sequence and Collaboration Diagrams
6. Draw relevant State Chart and Activity Diagrams for the same problem.
7. Implement the problem with the detailed design structure (1-6) and satisfy the SRS.
8. Test the developed software system.Identify the areas in which refinement is needed.
9. Refine the design of the modules using appropriate design patterns and implement the same.
10. Test the software again after refinement to check, whether it satisfies theSRS, and carry over the refinement if needed.

SUGGESTED DOMAINS FOR MINI PROJECT

1. Passport automation system.
2. Book bank
3. Exam Registration
4. Stock maintenance system.
5. Online course reservation system
6. E-ticketing
7. Software personnel management system
8. Credit card processing
9. e-book management system
10. Recruitment system
11. Foreign trading system
12. Conference Management System
13. BPO Management System
14. Library Management System
15. Student Information System

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Understand the requirements for a given problem specification. (K2)
- Designand draw UML diagrams. (K3)
- Map design to code and implement usingUML diagrams. (K4)
- Designand generate test cases. (K3)
- DesignUML diagrams by applying appropriate design patterns (K3).

LABORATORY REQUIREMENT FOR BATCH OF 38 STUDENTS

Hardware:

• Standalone Desktops - 38 Nos

Software:

• ArgoUML that supports UML 1.4 and higher

• Selenium, JUnit or Apache JMeter

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1701	DISTRIBUTED SYSTEMS	3	0	0	3

OBJECTIVES

- To introduce the computation and communication models of distributed systems
- To illustrate the issues of synchronization and collection of information in distributed systems
- To educate distributed mutual exclusion and distributed deadlock detection techniques
- To elucidate agreement protocols and Fault Tolerance mechanisms in Distributed Systems
- To explain the features of Peer-to-Peer systems and memory consistency models.

UNIT I INTRODUCTION 8

Introduction: Definition-Relation to computer system components -- Motivation -- Relation to parallel multiprocessor/multicomputer systems -- Message-passing systems versus shared memory systems -- Primitives for distributed communication -- Synchronous versus asynchronous executions -- Design issues and challenges; A model of distributed computations: A distributed program -- A model of distributed executions -- Models of communication networks – Global state of a distributed system -- Cuts of a distributed computation -- Past and future cones of an event -- Models of process communications.

UNIT II LOGICAL TIME AND GLOBAL STATE 10

Logical Time: Physical clock synchronization: NTP -- A framework for a system of logical clocks -- Scalar time -- Vector time; Message ordering and group communication: Message ordering paradigms -- Asynchronous execution with synchronous communication – Synchronous program order on an asynchronous system -- Group communication -- Causal order (CO) Total order; Global state and snapshot recording algorithms: Introduction -- System model and definitions – Snapshot algorithms for FIFO channels.

UNIT III DISTRIBUTED MUTEX AND DEADLOCK 10

Distributed mutual exclusion algorithms: Introduction – Preliminaries -- Lamport’s algorithm -- Ricart–Agrawala algorithm -- Quorum-based mutual exclusion algorithms -- Maekawa's algorithm -- Token-based algorithms -- Suzuki–Kasami’s broadcast algorithm; Deadlock detection in distributed systems: Introduction -- System model – Preliminaries -- Models of deadlocks -- Knapp’s classification of distributed deadlock detection algorithms -- Mitchell and Merritt’s algorithm for the single resource model -- Chandy–Misra–Haas algorithm for the AND model -- Chandy–Misra–Haas algorithm for the OR model.

UNIT IV CONSENSUS AND RECOVERY 10

Consensus and agreement algorithms: Problem definition -- Overview of results -- Agreement in a failure-free system (synchronous or asynchronous) -- Agreement in (message-passing) synchronous systems with failures; Check pointing and rollback recovery: Introduction -- Background and definitions -- Issues in failure recovery --Checkpoint-based recovery -- Log-based rollback recovery -- Koo–Toueg coordinated check pointing algorithm -- Juang–Venkatesan algorithm for asynchronous checkpointing and recovery.

UNIT V PEER TO PEER AND DISTRIBUTED SHARED MEMORY 7

Peer-to-peer computing and overlay graphs: Introduction – Data indexing and overlays -- Tapestry; Distributed shared memory: Abstraction and advantages -- Memory consistency models -- Lamport's Bakery Algorithm.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Realize the foundations of Distributed Systems (K2)
- Able to solve synchronization and state consistency problems (K3)
- Demonstrate the resource sharing techniques in Distributed systems (K3)
- Comprehend the working model of consensus and reliability of Distributed Systems (K3)
- Identify the fundamentals of Peer-to-Peer Systems (K2)

TEXTBOOKS

1. Kshemkalyani Ajay D, Mukesh Singhal.“Distributed computing: Principles, Algorithms and Systems”. Cambridge University Press, 2011.
2. Mukesh Singhal,Niranjan G Shivaratri. “Advanced Concepts in Operating Systems”. McGraw-Hill, 1994.

REFERENCE BOOKS

1. George Coulouris, Jean Dollimore, Tim Kindberg, “Distributed Systems Concepts and Design”, Fifth Edition,Pearson Education, 2012.
2. Pradeep K Sinha, “Distributed Operating Systems: Concepts and Design”, PrenticeHall of India,2007.
3. Tanenbaum A S, Van Steen M, “Distributed Systems: Principles and Paradigms”, Pearson Education, 2007.
4. Liu M L, “Distributed Computing, Principles and Applications”, PearsonEducation, 2004.
5. NancyA Lynch, “Distributed Algorithms”, Morgan Kaufman Publishers, USA, 2003.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1702	MOBILE COMPUTING	3	0	0	3

OBJECTIVES

- Understand the basic concepts of mobile computing
- Be familiar with the protocol stack
- Be exposed to Ad-Hoc networks
- Learn the basics of mobile telecommunication system
- Gain knowledge about different mobile platforms and application development environments.

UNIT I INTRODUCTION 9

Introduction to Mobile Computing; MediaAccess Control (MAC): Motivation for a specialized MAC– SDMA – FDMA – TDMA – CDMA– Comparison of S/T/F/CDMA; Wireless LAN: Infrastructure based and adhoc networks – IEEE802.11 – Bluetooth.

UNIT II NETWORK LAYER 9

MobileIP: Goals, assumptions andrequirements – Entitiesandterminology – IP packetdelivery – Agentdiscovery – Registration –Tunneling andencapsulation– Optimizations; DHCP;Adhoc:Routing – Proactive routing protocol- DSDV – Reactive routing protocol - DSR, AODV – Hybrid routing –ZRP – Multicast Routing- ODMRP; VANET.

UNIT III TRANSPORT AND APPLICATION LAYER 9

Mobile Transport Layer: Traditional TCP – ClassicalTCP improvements; Wireless Application Protocol(WAP): Architecture – WDP – WTLS – WTP –WSP – WAE – WML –WTA.

UNIT IV MOBILE TELECOMMUNICATION SYSTEM 9

Introduction; GlobalSystemforMobileCommunication (GSM): Mobileservices– Systemarchitecture – Radiointerface – Protocols– Localization and calling– Handover – Security;GeneralPacketRadioService (GPRS); Universal MobileTelecommunication System(UMTS): UMTS system architecture – UTRAN – Core network – Handover.

UNIT V MOBILE PLATFORMS AND APPLICATION ENVIRONMENTS 9

iOS: iOS Architecture Layers – iOS Simulator; Android: Platform architecture – Developing android applications – Anatomy of android applications – Android SDK; Mobile Web.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Identify the functionalities of various MAC protocols (K3)
- Explain the functionalities of mobile network layer and routing in Ad hoc networks (K3)
- Analyze the transport and application layer protocols (K3)
- Explain the basics of mobile telecommunication system (K2)
- Develop a mobile application (K3).

TEXTBOOKS

1. JochenH Schller, “Mobile Communications”, Pearson Education, New Delhi, 2nd Edition,2007 (Unit I–IV).
2. Helal, Sumi, Raja Bose, Wendong Li, “Mobile Platforms and DevelopmentEnvironments.” Synthesis Lectures on Mobile and Pervasive Computing,2012 (Unit V).

REFERENCE BOOKS

1. Helal,Abdelsalam A, et al.“ Any Time, Anywhere Computing: Mobile Computing Concepts and Technology”, Vol 522, Springer Science & Business Media, 1st Edition,1999.
2. Dharma Prakash Agarval, QingandAn Zeng, “Introduction to Wireless and Mobile systems”, Thomson Asia, 3rd Edition, 2005.
3. Uwe Hansmann, LotharMerk, MartinS Nicklons and Thomas Stober, “Principles of Mobile Computing”, Springer, 2003.
4. William C Y Lee, “Mobile Cellular Telecommunications–Analog and Digital Systems”, 2nd Edition, Tata Mc Graw Hill Edition,2006.
5. C K Toh, “AdHoc Mobile Wireless Networks”, Pearson Education, 1st Edition,2002.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1703	GRAPHICS AND MULTIMEDIA	3	0	0	3

OBJECTIVES

- To gain knowledge on different display devices and its working principles
- To understand the2D and3D dimensional graphics representation and object transformations
- To understand illumination principles andcolor models usedin output devices
- To understand basic concepts of multimedia
- To exploreBlendergraphics tool and design animations.

UNIT I DISPLAY SYSTEMS AND OUTPUT PRIMITIVES 8

Introduction to computer graphics – Applications; Overview of graphics systems:Videodisplay devices– Rasterscansystems – Random scansystems; Output primitives: Points and lines – Loading the frame buffer – Line drawing algorithms: DDA and Bresenham's line drawing algorithms – Circle and ellipse generating algorithms – Pixel addressing and object geometry.

UNIT II TWO-DIMENSIONAL GRAPHICS 9

Twodimensional geometric transformations:Basic transformations – Matrix representations andhomogeneous coordinates – Composite transformations; Two-dimensional viewing:Viewingpipeline – viewing coordinate reference frame – Window to viewport coordinate transformation – Clipping operations: Point and text clipping – Line and polygon clipping algorithms.

UNIT III THREE-DIMENSIONAL GRAPHICS 10

Threedimensional concepts; Threedimensional object representations:Polygonsurfaces– Polygon tables– Planeequations – Polygon meshes– Curved linesandsurfaces– Quadratic surfaces– Blobbyobjects– Splinerepresentations – Bezier curves and surfaces;Three Dimensional Geometric and Modeling Transformations: Translation – Rotation – Scaling – Composite transformations; Three Dimensional Viewing:Viewingpipeline – Viewingcoordinates – Projections – View volumes – Clipping.

UNIT IV ILLUMINATION MODELS AND ANIMATION 8

Lightsources– Basic illumination models:Ambient, Diffuse,Specular Components of the Phong model; Color Models: Properties of light – Standard primariesand chromaticity diagram – RGB, YIQ, CMY, HSV and HLS color models; Computer Animation: Designof animation sequences – Keyframe systems - Motionspecifications.

UNIT V MULTIMEDIA 10

Multimedia SystemsDesign:Multimedia elements – Multimedia applications – Multimedia systems architecture – Definingobjects for multimedia systems – Multimedia datainterface standards; Compression and decompression; Data and File Format Standards; Hypermedia Messaging; Case Study – Blender Graphics: Fundamentals – Drawing Basic Shapes– Modelling – Shading & textures.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Applythe algorithms to manipulate output primitives such as line, circle, ellipse (K3)
- Demonstrate transformations, representations and clipping on 2D objects and map window to viewport transformations (K3)
- ApplythreeDimensional concepts like representations, geometric transformations, and projections (K3)
- Understand the working of different illumination andcolor models used to render an animation scene (K2)
- Understand different typesof multimedia file formats, compression techniques and design basic 3D Scenes usingBlender(K2).

TEXTBOOKS

1. Donald Hearn, Pauline BakerM, "Computer Graphics", PrenticeHall, New Delhi, 2007.
2. Andleigh P K, Kiran Thakrar, "Multimedia SystemsandDesign", PHI, 2003.

REFERENCE BOOKS

1. Foley, Vandam, Feiner, Hughes, "Computer Graphics: Principles and Practice", 2nd Edition,Pearson Education, 2003.
2. Jeffrey McConnell, "Computer Graphics: Theory into Practice", Jones and Bartlett Publishers, 2006.
3. Hill F S Jr, "Computer Graphics", MaxwellMacmillan, 1990.
4. Peter Shirley, Michael Ashikhmin, Michael Gleicher, Stephen R Marschner, Erik Reinhard, KelvinSung, AK Peters, "Fundamentals of Computer Graphics",CRC Press, 2010.
5. https://www.blender.org/support/tutorials/

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1704	MANAGEMENT AND ETHICAL PRACTICES	3	0	0	3

OBJECTIVES

- To train basic and applied fields of Management
- To improve the Managerial skills
- To prepare the students to the management world
- To create an awareness on Engineering Ethics and Human Values.
- To learn technical, legal, and ethical issues involved in computer security

UNIT I OVERVIEW OF MANAGEMENT & PLANNING 9

Organization – Role of managers – Evolution of management thought – Managing globally– Strategies for international business. Planning process– Types of plans – Decision Making – Types of decision– Decision making Process – Rationaldecision-making process– Decision making under different conditions.

UNIT II ORGANIZING AND DIRECTING 9

Nature and purpose of organizing – Line and staff authority – Departmentation – Centralization anddecentralization – Delegation of authority – Staffing – Selection and recruitment – Orientation – Performance appraisal; Motivation and Satisfaction – Motivation theories leadership – Leadership theories – Hurdles to effective communication.

UNIT III ETHICS IN ENGINEERING 9

Senses of engineering ethics – Variety of moral issues – Types of inquiry – Moral dilemmas – Moral autonomy – Kohlberg'stheory – Gilligan's theory – Consensus and controversy – Models of professional roles - Theoriesaboutright action – Self-interest – Customs andreligion – Uses of ethicaltheories - Ethics of AIand Machine Learning.

UNIT IV SAFETY, RESPONSIBILITIES AND RIGHTS 9

Safety andRisk: Assessment of safetyandrisk – Risk benefitAnalysis andreducing risk-Respectfor authority – Collectivebargaining – Confidentiality– Conflictsof interest – Occupational crime– Professional rights– Employee rights– Intellectual Property Rights (IPR) – Discrimination.

UNIT V CYBERSECURITY ETHICS 9

Introduction – Controlling AccessFlow– Protecting Privacy– Dealingwith Intrusion – Managing Distributed Resources – Encouraging Exploration – FosteringResponsibility – Asserting Ownership – Three Ethical Frameworks

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Describe basic and applied fields of Management (K2)
- Describe and practice Managerial skills (K3)
- Describe and practice Engineering Ethics and Human Values (K3)

- Describe and use safety, responsibility, and rights (K3)
- Describe ethical issues in cyber security (K2)

TEXTBOOKS

1. Hellriegel, Slocum, Jackson, “Management -A Competency Based Approach”, Thomson South Western, 10th edition, 2007.
2. Govindarajan M, Natarajan S, SenthilKumar V S, “Engineering Ethics”, PrenticeHall of India,New Delhi, 2004.

REFERENCE BOOKS

1. Harold Koontz, Heinz Weihrich, Mark V Cannice, “Management – Aglobal & Entrepreneurial Perspective”, Tata McGrawHill, 12th edition, 2007.
2. Stephen P Robbins,MaryCoulter, “Management”, PrenticeHallof India,8thedition.
3. MikeW Martin, Roland Schinzinger, “Ethics in Engineering”, Tata McGraw Hill, New Delhi, 2003.
4. Mary Manjikian, "Cyber security Ethics an Introduction", Routledge Taylor & Francis Group, 2018.
5. George Ledin, "Computer Security, Ethics and Society", McGraw-Hill, 2010.
6. http://www.cs.bath.ac.uk/~jjb/web/ai.html

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1711	MOBILE APPLICATION DEVELOPMENT LAB	0	0	3	1.5

OBJECTIVES

- To develop nativemobile applications
- To develop hybrid mobile applications.

SUGGESTIVE EXPERIMENTS

1. Develop an application that uses GUI components, LayoutManagers and event listeners.
2. Develop an application to simulate a keyboard.
3. Createan application that uses graphical primitives.
4. Develop an application that makesuse of databases.
5. Implement an application that uses Multi-threading.
6. Develop a nativeapplication that uses GPS locationinformation.
7. Implement an application that writesdatato the SD card.
8. Implement an application thatsenda SMS andcreatesan alertupon receiving the SMS.
9. Createan application that makesuse of Menu.
10. Develop an application to buildan alarmclock.
11. Implement a hybrid mobile application for displaying a website.
12. Mini Project (Food delivery app,Attendance tracking app,Onlineticket booking app etc.)

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Develop mobile applications usingGUI, Layouts and Event Listener(K3)
- Develop mobile applications using Graphical primitives, Databases, Multithreading and GPS(K3)
- Develop mobile applications usingSD Card, SMS and Notification(K3)
- Develop hybrid mobile applications(K3)
- Develop a mobile app for simpleneeds(K3).

LABORATORY REQUIREMENT FOR BATCH OF 38 STUDENTS

Hardware:

1. Standalone Desktops preferably Intel i5 Machines - 38 Nos

Software:

1. Android oriOS orEquivalent MobileApplication Development Tools with appropriate emulators and debuggers.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1712	GRAPHICS AND MULTIMEDIA LAB	0	0	3	1.5

OBJECTIVES

- Understand graphics programming in OpenGL using OpenGL and GLUT frameworks
- Implement algorithms for line and circle drawing
- Apply2D, 3D transformations and clipping on objects
- Learn to draw 3D objects and apply projection techniques
- Explore image manipulation and enhancement techniques
- Create3D animation usingany tool.

SUGGESTIVE EXPERIMENTS

1. Studyof Basic output primitives in OpenGL
2. Implementation of Algorithms for drawing 2D Primitives –
    a. Line (DDA, Bresenham's) - all slopes
    b. Circle (Midpoint)
3. 2D Geometric transformations – Translation, Rotation,Scaling, Reflection and Shear
4. 2D Composite Transformations and Window to viewport mapping
5. Implementation of Line clipping algorithm
6. 3D Geometric Transformations - Translation, Rotation and Scaling
7. 3D Projections - Paralleland Perspective projection
8. Image Editing and Manipulation -
    c. Basic Operations on imagelike applying masks,filters,adding/re- moving noise
    d. Creating gif animated images
9. Creation of a simple2D animation
10. Creation of a simple3D animation

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Demonstrate drawing of basicoutput primitives (lineandcircle) using algorithms and hence draw complexshapes usingthem(K3)
- Illustrate basic, composite transformations and clipping on 2 dimensional objects (K2)
- Applytransformations andprojections on 3 dimensional objects anddevelop any scene with features including lighting, textures, shadows, changing cameraangles(K3)
- Applybasic operations on images,createGIF animated imagesand2D animation sequence (K3)
- Develop a simple3D animation (K3)

LABORATORY REQUIREMENT FOR BATCH OF 38 STUDENTS

Hardware:

1. Standalone Desktop Machines - 38 Nos

Software:

1. C/C++/Java
2. OpenGL/GIMP
3. Blender/Maya

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1818	PROJECT WORK	0	0	12	6

OBJECTIVES

- To develop the ability to solve a specific problem right from its identification and literature review till the successful solution of the same.
- To train the students in preparing project reports and to face reviews and viva voce examination.

The students in a group of 3 to 4 works on a topic approved by the head of the department under the guidance of a faculty member and prepares a comprehensive project report after completing the work to the satisfaction of the supervisor. The progress of the project is evaluated based on a minimum of three reviews. The review committee may be constituted by the Head of the Department. A project report is required at the end of the semester. The project work is evaluated based on oral presentation and the project report jointly by external and internal examiners constituted by the Head of the Department.

TOTAL PERIODS: 180

OUTCOMES

On successful completion of this course, the student will be able to

- On Completion of the project work students will be able to take up any challenging practical problems and find solution by formulating proper methodology.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1521	ADVANCED DATABASE MANAGEMENT SYSTEMS	3	0	0	3

OBJECTIVES

- To acquire knowledge on parallel and distributed databases and their ap- plications
- To study the usage and applications of Object and Object Relational Databases
- To learn aboutXML Database
- To acquire Knowledge in Unstructured Databases
- To understand the concepts of HBase and HIVE.

UNIT I PARALLEL AND DISTRIBUTED DATABASES 9

Database System Architectures: Centralized and client-server architectures -- Server system architectures -- Parallel systems -- Distributed systems; Parallel Databases: I/O parallelism -- Inter and intra query parallelism -- Inter and intra operation parallelism -- Design of parallel systems -- Distributed database concepts -- Distributed data storage -- Distributed transactions -- Commit protocols -- Concurrency control -- Distributed query processing -- Case studies.

UNIT II OBJECT AND OBJECT RELATIONAL DATABASES 9

Concepts for Object Databases: Object identity -- Object structure -- Type constructors -- Encapsulation of operations -- Methods -- Persistence -- Type and class hierarchies -- Inheritance -- Complex Objects -- Object database standards; Languages and Design: ODMG model -- ODL -- OQL -- Object relational and extended; Relational Systems: Object relational features in SQL/Oracle -- Case studies.

UNIT III XML DATEBASES 9

XML Databases: XML-related technologies – XML Schema -- XML Query Languages -- Storing XML in databases -- XML and SQL -- Native XML databases -- Web databases.

UNIT IV NOSQL DATABASE 9

Why NoSQL: Aggregate data models -- The CAP theorem -- Key-value databases -- Document databases -- Column-Family stores -- Graph databases.

UNIT V HBASE ANDHIVE 9

Introduction to Hadoop and MapReduce; HBase Basic Features: CRUD operations -- Batch operations -- Row Locks; Advanced Features: Filters -- Counters –Htablepool; HiveQL: Data definition -- Data manipulation -- Queries.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- DesignParalleland Distributed Databases (K3)
- Understand Object and Object Relational Databases (K2)
- Designan application usingXML Database (K3)
- Implement different NoSQL Databases (K3)
- Designan application usingthe concepts of HBase and HIVE (K3).

TEXTBOOKS

1. Elmasri, Ramez, Shamkant Navathe, “Fundamentals of database systems”, Addison-Wesley Publishing Company, 2011 (Units I, II, III)
2. Fowler,Martin,Pramod Sadalage, “NoSQL Distilled” Addison Wesely,2013 (Unit IV).
3. George,Lars, “HBase: The Definitive Guide: Random Accessto Your Planet-Size Data”,OReilly MediaInc, 2011 (Unit V).

REFERENCE BOOKS

1. Capriolo, Edward, Dean Wampler, Jason Rutherglen. “Programming Hive: Data Warehouse andQueryLanguage for Hadoop”, O’Reilly MediaInc,2012.
2. Henry F Korth,Abraham Silberschatz, S Sudharshan, “Database SystemConcepts”, 6th Edition,McGrawHill, 2011.
3. CarloZaniolo, StefanoCeri, Christos Faloutsos, RichardT Snodgrass, V S Subrahmanian, Roberto Zicari, “Advanced Database Systems”, Morgan Kaufmann publishers, 2006
4. Membrey, Peter, EelcoPlugge, DUP TimHawkins. “The Definitive Guide to MongoDB: The NoSQL Database for Cloudand Desktop Computing”, Apress,2011.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1522	SOFTWARE TESTING	3	0	0	3

OBJECTIVES

- To be familiar with objectives, principles, process, and factors of software testing
- To learn the design of test cases using Black box approach and white box approaches
- To understand and apply the levels of testing
- To understand automation methods in software testing
- To test the product to check the product Quality.

UNIT I TESTING FUNDAMENTALS 9

Objectives and Principles: Fundamental test process -- Test levels -- Establishing a testing policy -- Structured approach to testing -- Test factors -- Developing risk matrix -- Steps in software testing process.

UNIT II TEST CASE DESIGN STRATEGIES AND METHODS 9

Strategies and Methods for Test Case Design: Introduction to testing design strategies -- Test case design strategies -- Using the black box approach to test case design -- Random testing -- Equivalence class partitioning -- Boundary value analysis -- Cause-and-effect graphing -- State transition testing -- Error guessing; Using the white box approach to test design -- Coverage and control flow graphs -- Covering code logic -- Paths -- Data flow and white box test design -- Loop testing--Mutation testing;

UNIT III TESTING FUNDAMENTALS 9

Levels of Testing: The Need for levels of testing -- Unit test functions, procedures, classes and methods as units -- Unit test planning -- Designing the unit tests -- The test harness -- Running the unit tests and recording results -- Integration tests -- Designing integration tests -- Integration test planning -- System test -- Functional testing -- Performance testing-- Stress testing -- Configuration testing -- Security testing -- Recovery testing -- Regression testing -- Alpha, beta and acceptance tests.

UNIT IV TEST AUTOMATION AND MANAGEMENT 9

Test Planning, Management, execution and Reporting: Test planning -- Management -- Process -- Reporting; Software Test Automation: Testing tools; Controlling and Monitoring the Testing Process: Measurements and milestones for controlling and monitoring -- Criteria for test completion -- Configuration management -- Controlling and monitoring.

UNIT V SOFTWARE QUALITY ASSURANCE 9

Software Quality; Quality Factors: Product operation, revision and transition; Components of SQA: System and architecture; Pre-Project Components; Contract Review; Development and Quality Plans; SQA Components in Project Life Cycle: SQA defect removal policies; Reviews; Project progress control; Costs; Quality Management Standards; Project Process Standards; Management and its Role in SQA; SQA Unit.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Understand software testing objectives, principles and practices (K2)
- Apply Black and white box approach to design test cases (K3)
- Understand and apply levels of testing (K3)
- Apply Automation tools for software testing in projects (K3)
- Analyze the product Quality (K3).

TEXTBOOKS

1. Gopalaswamy Ramesh, Srinivasan Desikan, "Software Testing: Principles and Practices", Pearson Education, New Delhi, 2006 (Units I, IV).
2. ELlene Burnstein," Practical Software Testing", Springer International Edition, Chennai,
1. 2003 (Units II, III, IV).

REFERENCE BOOKS

1. Aditya Mathur, "Foundations of SoftwareTesting", Pearson Education,2008.
2. DanielGalin, "Software Quality Assurance – From Theoryto Implementation", Pearson Education, 2009 (UNIT-V)
3. AlanC Gillies, "Software Quality Theory and Management", 2nd Edition, Cengage Learning, 2003.
4. Robert Furtell, Donald Shafer, and Linda Shafer, "Quality Software Project Management", Pearson Education Asia, 2002.
5. Ron Patton,"Software Testing", 2nd Edition,Pearson Education, 2007.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1523	DIGITAL SIGNAL PROCESSING	3	0	0	3

OBJECTIVES

- To review discrete-time signals and systems
- To introduce DFT and its properties for discrete-time signal analysis
- To teach the design of IIR and FIR filters for filtering the undesired signals
- To introduce the DSP for engineering applications.

UNIT I REVIEW OF SIGNALS, SYSTEMS AND SIGNAL PROCESSING 9

Basic elements of DSP system, Discrete-Time (DT) signals -- Classification of DT signals, Concepts of frequency in Analog and Digital Signals, Need for sampling and sampling theorem, Discrete -- Time systems -- Types of DT system, Analysis of discrete time LTI systems, Linear convolution, Z transform and its applications for LTI system analysis.

UNIT II FREQUENCY ANALYSIS OF DISCRETE-TIME SIGNALS 9

Introduction to Discrete Fourier Transform (DFT) -- Properties of DFT -- Circular convolution, Linear filtering based on DFT, Filtering of long data sequences using Overlap -- save method and Overlap -- add method, FFT Algorithms -- Decimation-in-time (DIT) FFT Algorithm, Decimation-in-frequency (DIF) FFT Algorithm.

UNIT III DESIGN OF IIR FILTER 9

Design of analog IIR filter – Butterworth and Chebyshev IIR filters, Design of discrete time IIR filter from analog filter -- IIR filter design by Impulse Invariance technique (IIT) and Bilinear transformation technique (BLT), Structures of IIR filter -- Direct form-I, Direct form-II, Cascade and Parallel form, Finite word length effects of IIR filter -- limit cycle oscillations, dead band, signal scaling.

UNIT IV DESIGN OF FIR FILTER 9

Linear phase FIR filter, FIR Filter design using Rectangular, Hamming and Hanning windowing techniques, Structures of FIR filter -- Transversal, linear-phase and poly-phase structures, Finite word length effects of FIR filter.

UNIT V APPLICATIONS OF DIGITAL SIGNAL PROCESSING 9

Multi-rate signal processing - Decimation, Interpolation, Multi-stage design of Decimation and Interpolation filters, Sub-band coding of speech signals, Adaptive filtering – Noise cancellation, Speech enhancement.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

1. Understand signals, systems and basics of digital signal processing (K2)
2. Understand and apply DFT and FFT for the analysis of DT signals & systems (K3)
3. Design and realize IIR filters using impulse invariant and bilinear transformation techniques (K3)
4. Design and realize FIR filters using various window techniques (K3)

5. Understand the need of DSP for engineering applications (K2).

TEXTBOOKS

1. John G Proakis,Dimitris G Manolakis, "Digital Signal Processing – Principles, Algorithms and Applications", 4th Edition,Pearson Education, Prentice Hall, 2007.
2. Sanjit K Mitra, "Digital Signal Processing – A Computer Based Approach",3rd EditionTata McGrawHill, 2007.

REFERENCE BOOKS

1. S Salivahanan, "DigitalSignalProcessing", ThirdEdition,McGrawHill Education (India) Private Limited, 2015.
2. V Oppenheim, R W Schafer, J R Buck, "Discrete-Time Signal Processing",8th Indian Reprint, Pearson 2004.
3. Emmanuel C Ifeachor, Barrie W Jervis, "Digital Signal Processing", 2nd Edition, Pearson Education, PrenticeHall 2002.
4. Andreas Antoniou, "DigitalSignal Processing", Tata McGrawHill, (a) 5. P RameshBabu, "DigitalSignal Processing", 4th Edition,Scitech Publications (India) Pvt Ltd, 2011.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1524	LOGIC PROGRAMMING	3	0	0	3

OBJECTIVES

- To understand the foundations of Logic programming
- To learn programming in PROLOG
- To implement informed and uninformed search algorithms in PROLOG
- To implement Expert system shell in PROLOG.

UNIT I INTRODUCTION TO LOGIC 9

Foundations of Propositional Logic: Syntax and semantics -- Semantic entailment; Normal Forms; Resolution in PL; Horn Logic; Foundations of First Order Logic: Syntax and semantics; Normal Forms; Undecidability of Predicate Logic; Resolution in FOL: Unification; Refinements of resolution: P-resolution -- N-resolution -- Linear resolution -- Unit resolution -- SLD resolution – LUSH resolution.

UNIT II INTRODUCTION TO LOGIC PROGRAMMING 9

Foundations: Answer generation; Horn Clause Programs: Semantics of logic program -- Procedural semantics -- Model-theoretic semantics; Evaluation Strategies: Swapping lemma -- PROLOG's evaluation strategy -- DFS -- BFS.

UNIT III PROGRAMMING IN PROLOG 9

Syntax & Semantics of PROLOG -- Programming with relations -- Facts -- Questions -- Variables -- Conjunctions of goals -- Backtracking -- Rules -- Structures -- Recursive programs -- Lists -- Controlling execution -- The cut -- Input output -- Exception handling.

UNIT IV PROLOG AND ARTIFICIAL INTELLIGENCE 9

Overview of PROLOG: Lists -- Operators -- Arithmetic -- Data structures -- Eight Queens Problem; Operations on Data Structures: Representing and sorting lists -- Representing sets by binary trees -- Insertion & deletion in binary dictionary -- Displaying trees -- Graphs; Advanced Tree Representations; Problem Solving Strategies in AI: DFS -- BFS -- Best-first search -- A* Search; Problem Reduction & AND/OR Graph: Basic AND/OR search procedure -- Best-first AND/OR search; Game Playing: The minimax principle -- The alpha-beta algorithm -- Minimax-based programs.

UNIT V PROLOG AND EXPERT SYSTEMS 9

Foundations: Introduction to expert systems -- Features of expert systems -- Functions of an expert system -- Structure of an expert system; If-then rules for representing knowledge; Developing a Simple Shell: Implementation -- Dealing with uncertainty;

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Understand the foundations of logic (K2)
- Understand the foundations of logic programming (K2)
- Write programs in PROLOG (K3)

- Implement AI search algorithms in PROLOG (K3)
- Implement a simple Expert system shell in PROLOG (K3).

TEXTBOOKS

1. Uwe Schoning, “Logic for Computer Scientists”, Birkhauser, 1999 (Units I, II).
2. Ivan Bratko, “PROLOG: Programming for Artificial Intelligence”, 4th Edition, Pearson, 2011 (Units III, IV, V).

REFERENCE BOOKS

1. Kees Doets, “From Logic to Logic Programming”, MIT Press 1994.
2. Patrick Blackburn, Johan Bos, Kristina Streignitz, “Learn PROLOG Now”, College Publications, 2006.
3. Dennis Merritt, “Building Expert Systemsin PROLOG”, Amzi! Inc. 2000
4. Helder Coelho, Jose C Cotta, “PROLOG by Example:How to Learn, Teach and Use It”, Springer–Verlag, 2011.
5. W F Clocksin,C S Mellish, “Programming in PROLOG”, Springer-Verlag,2016.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1525	WIRELESS AD HOC AND SENSOR NETWORKS	3	0	0	3

OBJECTIVES

- To learn about the issues and challenges in the design of wireless ad hoc networks.
- To understand the working of MAC Protocols for ad hoc and sensor networks
- To understand the working of Routing Protocols for ad hoc and sensor networks
- To learn about the Transport Layer protocols and their QoS for adhoc and sensor networks
- To understand various security issues in ad hoc and sensor networks and the corresponding solutions.

UNIT I MAC AND ROUTING IN ADHOC NETWORKS 9

Introduction -- Issues and challenges in ad hoc networks -- MAC Layer Protocols for wireless ad hoc networks -- Contention-Based MAC protocols -- MAC Protocols using Directional Antennas -- Multiple-Channel MAC Protocols -- Power-Aware MAC Protocols -- Routing in Adhoc Networks -- Design Issues -- Proactive, Reactive and Hybrid Routing Protocols.

UNIT II TRANSPORT AND QOS IN AD HOC NETWORKS 9

TCP's challenges and Design Issues in Ad Hoc Networks -- Transport protocols for ad hoc networks -- Issues and Challenges in providing QoS -- MAC Layer QoS solutions -- Network Layer QoS solutions -- QoS Model.

UNIT III MAC AND ROUTING IN WIRELESS SENSOR NETWORKS 9

Introduction -- Applications -- Challenges -- Sensor network architecture -- MAC Protocols for wireless sensor networks -- Low duty cycle protocols and wakeup concepts -- Contention Based protocols -- Schedule-Based protocols -- IEEE 802.15.4 Zigbee -- Topology Control -- Routing Protocols.

UNIT IV TRANSPORT AND QOS IN WIRELESS SENSOR NETWORKS 9

Data-Centric and Contention-Based Networking -- Transport Layer and QoS in Wireless Sensor Networks -- Congestion Control in network processing -- Operating systems for wireless sensor networks -- Examples.

UNIT V SECURITY IN ADHOC AND SENSOR NETWORKS 9

Security Attacks -- Key Distribution and Management -- Intrusion Detection -- Software based Anti-tamper techniques -- Water marking techniques -- Defense against routing attacks -- Secure Ad hoc routing protocols -- Broadcast authentication WSN protocols -- TESLA -- Biba -- Sensor Network Security Protocols -- SPINS.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Identify different issues in wireless ad hoc and sensor networks (K2)
- Analyze the MAC protocols developed for ad hoc and sensor networks (K2)

- Understand the working of routing protocols for ad hoc and sensor networks (K2)
- Analyse Transport Layer protocols and their QoS and to apply them for adhoc and sensor networks (K3)
- Identify and understand security issues in ad hoc and sensor networks (K2).

TEXTBOOKS

1. C Siva Ram Murthy, B S Manoj, "AdHoc Wireless Networks – Architectures and Protocols", Pearson Education, 2006.
2. Holger Karl, Andreas Willing, "Protocols and Architectures for Wireless Sensor Networks", John Wiley & Sons, 2005.

REFERENCE BOOKS

1. Subir Kumar Sarkar,T G Basavaraju, C Puttamadappa, "AdHoc Mobile WirelessNetworks", Auerbach Publications, 2008.
2. Carlos De Morais Cordeiro, Dharma Prakash Agrawal, "Ad Hoc and Sensor Networks: TheoryandApplications (2nd Edition)", World Scientific Publishing, 2011.
3. WaltenegusDargie, Christian Poellabauer, "Fundamentals of Wireless Sensor Networks Theoryand Practice",John Wiley and Sons, 2010.
4. Xiang-YangLi, "Wireless Ad Hoc and Sensor Networks: Theory and Applications", Cambridge university Press, 2008.

Course Code	Course Title	L	T	P	C
UCS1526	PROGRAMMING PARADIGMS	3	0	0	3

OBJECTIVES

- To introduce various programming paradigms
- To understand programming paradigms such as imperative, object-oriented, logic, functional and concurrent with illustrative examples.

UNIT I IMPERATIVE PROGRAMMING 9

Roleof programming languages; Programming paradigms;Structured programming: Data representation – Procedure activations – Examples in C.

UNIT II OBJECT ORIENTED PROGRAMMING 9

Abstract datatypes– objectmodel– object-oriented thinking – Examples inJava/Python/C++.

UNIT III FUNCTIONAL PROGRAMMING 9

Elements of functional programming: Valuesandoperations – Approaches to expression evaluation – Lexical scope – Type checking;Haskell:Expressions – Lists – Types and values– Control flow – Functions – Tuples– Examples.

UNIT IV LOGIC PROGRAMMING 9

Logic andHornclauses,Prolog:Data structures – Programming techniques – Control – Cuts – Examples.

UNIT V CONCURRENT PROGRAMMING 9

Concurrency concepts – Synchronization strategies – Interleaving – Liveness– Safe access to shared data– Examples in Java.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Understand imperative programming paradigm and some examples in C (K3)
- Understand object-oriented programming paradigm andexploresome examples in C++ or Java or Python (K3)
- Understand functional programming paradigm andexploresome examples in Haskell (K3)
- Understand logic programming paradigm and exploresome examples inProlog (K3)
- Understand concurrent programming paradigm and explore some examples in Java/Ada (K3).

TEXTBOOKS

1. Ravi Sethi,Viswanatha K V, “Programming Languages:Concepts and constructs”, 2nd Edition,Pearson, 2011
2. Allan B Tucker,RobertE Noonan, “Programming Languages: Principles and Paradigms”, 2nd Edition,McGrawHill, 2007

REFERENCE BOOKS

1. MichaelScott, “Programming Language Pragmatics”, 4th Edition,Morgan Kaufmann, 2015
2. RobertW Sebesta,“Concepts of Programming Languages”, 11th Edition, Pearson, 2016
3. Maurizio Gabbrielli, SimoneMartini, “Programming Languages: Principles and Paradigms”, Springer, 2010

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1621	ADVANCED JAVA PROGRAMMING	3	0	0	3

OBJECTIVES

- To understand the concepts of multithreading and collections
- To develop applications with JDBC
- To develop applications using Hibernate and struts
- To develop web applications using spring MVC framework

UNIT I MULTITHREADING AND COLLECTIONS 9

Introduction: Life cycle of a thread, Creating threads -- Synchronization: Code blocks, methods, classes -- Concurrent programming -- Introduction to collection framework -- Collection interfaces: Map, Queue, List, Set -- Collection classes: Abstract Collection, Abstract List, ArrayList, LinkedList, Generic class, HashSet, HashMap, TreeSet, TreeMap -- Legacy classes and interfaces.

UNIT II JAVA DATABASE CONNECTIVITY 9

Introduction: Components, specification, architecture -- JDBC drivers -- JDBC APIs -JDBC classes and interfaces -- JDBC process with SQL package -- Working with transactions.

UNITIII HIBERNATE ORM 9

Introduction: Architecture of hibernate-HQL -- Hibernate OR mapping -- developing hibernate application.

UNITIV JAVA STRUTS2 FRAMEWORK 9

Introduction to struts2 -- actions in struts -- dependency injection and inversion of control -- interceptors -- OGNL in struts2 -- struts2 tags -- validation in struts2.

UNIT V SPRING MVC FRAMEWORK 9

Overview of spring -- dependency injection -- spring libraries - developing spring application -- spring configuration -- Spring web MVC: Spring MVC architecture -- Components of MVC -- request mapping -- developing spring MVC application -- custom binding -- session attributes -- connecting MVC applications to the business layer.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Develop Java applications using multithreading (K3)
- Develop Java applications with Java Database connectivity (K3)
- Develop Hibernate applications using object relational mapping (K3)
- Develop Java enterprise edition web applications using struts framework (K3)
- Build web applications using spring MVC framework (K3).

TEXTBOOKS

1. “Core and Advanced Java, Black Book”, Dream tech press, 2018
2. Vaskaran Sarcar, “Java Design Patterns”, Apress,2016

REFERENCE BOOKS

1. Herbert Schildt, "Java: The Complete Reference", 11th Edition, McGraw Hill education, 2018.
2. Madhu Sudan Konda, "Just Hibernate – A Lightweight Introduction to the Hibernate Framework", O'Reilly Media,2014.
3. James Holmes, "Struts 2: The Complete Reference", McGraw Hill Education, 2008.
4. Amudhan G, "Spring MVC Beginner 's Guide", Packt Publishing, 2014.
5. Mark Grand, "Patterns in Java: A Catalog of Reusable Design Patterns Illustrated with UML", 2nd Edition, Wiley publishing.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1622	SOFTWARE PROJECT MANAGEMENT	3	0	0	3

OBJECTIVES

- To learn the key aspects of managerial process in software organization.
- To apply the practices followed in project planning, estimation, and scheduling
- To be familiar with the overall project activities in Agile and DevOps
- To study the various issues in people management
- To distinguish between security and risk activities related with software development.

UNIT I PROJECT PLANNING 9

Project Management: Project management life cycle -- Project evaluation and programme management -- Project planning; Process Models: Waterfall -- Spiral -- Prototyping -- Incremental delivery -- DSDM -- Agile -- Scrum --XP; Software Development: Lean software development -- Process model selection.

UNIT II ESTIMATION AND SCHEDULING 9

Effort Estimation Techniques: Expert judgement -- FPA -- FP mark II -- COSMIC -- COCOMO II -- Cost estimation; Scheduling: Project schedules -- Network model -- Identifying critical activities.

UNIT III AGILE ANDDEVOPS 9

Scrum: Organization -- Estimation -- Planning -- Changes in scrum -- Risk in scrum -- Product backlog -- Sprint backlog -- Release -- Identifying stories and preparing to build; DevOps: Motivation -- Cloud as a platform -- Operations -- Deployment pipeline.

UNIT IV SECURITY AND RISK MANAGEMENT 9

Cloud Security: Cloud information security objectives -- Cloud security services -- Secure development practices; Risk Management: Risk culture -- Risk managementprocess -- Risk attributes -- Risk identification -- Types of risk analysis -- Responding to risk -- Risk tracking -- Risk models.

UNIT V MONITORING AND ORGANIZING TEAMS 9

Visualizing Progress: Relationship between people and effort -- Task set and network -- Earned value analysis -- SCM -- Managing contracts -- Working in teams -- Developing metrics on a software development program -- Evaluating performance.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Select a process model for software development (K2)
- Estimate cost to manage the software development (K3)
- Develop a project using agile and devops software development practices (K3)
- Identify the activities of security and risk in various types of software projects (K1)

- Recognize management skills and techniques to develop commercial software projects (K2).

TEXTBOOKS

1. Bob Hughes, Mike Cotterell, Rajib Mall, "Software Project Management", 6th Edition,Tata McGrawHill, 2018.
2. Ravindranath Pandian, "Applied Software Risk Management A Guide for SoftwareProject Managers", Auerbach Publication, 2007.

REFERENCE BOOKS

1. Roger S Pressman, "Software Engineering – A Practitioner 's Approach", 7th Edition, McGraw-Hill International Edition,2010.
2. Rajib Mall, "Fundamentals of Software Engineering", 3rdedition, PHI Learning Pvt. Ltd., 2009.
3. Tridibesh Satpathy, "A Guideto the Scrum Body of Knowledge (SBOK Guide)", 2016 edition, SCRUMstudy, Phoenix,Arizona 85008 USA.
4. Len Bass, Ingo Weber and Liming Zhu, "DevOps: A Software Architect's Perspective", Pearson Education, 2016.
5. Ronald L Krutz, Russell Dean Vines, "Cloud Security– A comprehensiveGuideto Secure CloudComputing", Wiley-India, 2010.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1623	IMAGE PROCESSING AND ANALYSIS	3	0	0	3

OBJECTIVES

- To be familiar with digital image fundamentals
- To understand the basics of simple image enhancement techniques in Spatial and Frequency domain
- To learn the concepts of degradation function and restoration techniques
- To study image segmentation and representation techniques
- To be familiar with object recognition methods.

UNIT I DIGITAL IMAGE FUNDAMENTALS 9

Introduction to Digital Image Processing: Components -- Image sensing and acquisition -- Image sampling and quantization; Relationships between Pixels; 2D Mathematical Preliminaries: Array vs matrix operations -- Linear vs nonlinear operations -- Arithmetic -- Logical -- Statistical -- Spatial operations.

UNIT II IMAGE ENHANCEMENT 9

Spatial Domain: Gray level transformations -- Histogram processing; Basics of Spatial Filtering: Smoothing and sharpening spatial filtering; Frequency Domain: Introduction to fourier transform -- Smoothing and Sharpening Frequency Domain Filters: Ideal -- Butterworth -- Gaussian filters; Homomorphic Filtering.

UNIT III IMAGE RESTORATION 9

Image Restoration: Degradation model; Noise models; Restoration in the Presence of Noise using Spatial Filtering: Mean filters -- Order statistics -- Adaptive filters; Periodic Noise Reduction by Frequency Domain Filtering: Band reject filters -- Band pass filters -- Notch filters -- Optimum notch filtering; Inverse filtering; Wiener filtering.

UNIT IV IMAGE SEGMENTATION 9

Edge Detection: Edge linking via Hough transform; Thresholding: Intensity thresholding -- Otsu's thresholding; Region Based Segmentation: Region growing -- Region splitting and merging; Morphological Processing: Erosion and dilation; Segmentation by Morphological Watersheds: Basic concepts -- Dam construction -- Watershed segmentation algorithm.

UNIT V FEATURE ANALYSIS AND OBJECT RECOGNITION 9

Boundary representation; Boundary Descriptors: Fourier descriptor; Regional descriptors -- Texture; Relational descriptors; Object Recognition: Patterns and pattern classes; Recognition based on decision theoretic methods; Case Study for Image Recognition: Face recognition.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Understand the basics and fundamentals of digital image processing, such as digitization, sampling, quantization (K2)

- Operate on images using the techniques of smoothing, sharpening and enhancement in both spatial and frequency domain (K3)
- Understand the restoration concepts and filtering techniques (K2)
- Analyse the segmentation methods and apply in real time applications (K3)
- Understand and analyse the feature extraction and object recognition methods (K3).

TEXTBOOKS

1. Rafael C Gonzalez, RichardE Woods,"DigitalImageProcessing", Pearson, 3rdEdition,2010.
2. Anil K Jain, "Fundamentals of Digital Image Processing", Pearson, 2002.

REFERENCE BOOKS

1. Kenneth R Castleman, "DigitalImage Processing", Pearson, 2006.
2. Rafael C Gonzalez, Richard E Woods , Steven Eddins, "Digital Image Processing Using MATLAB", Pearson Education, Inc., 2011.
3. S Sridhar, "DigitalImageProcessing", 2ndEdition,OxfordUniversity,2016.
4. William K Pratt, "DigitalImage Processing", John Wiley, New York, 2002.
5. MilanSonka, Roger Boyle, Vaclav Hlavac, "Image Processing, Analysis and Machine Vision", Brookes/Cole, Vikas Publishing House, 2ndedition, 1999.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1624	INTERNET OF THINGS	3	0	0	3

OBJECTIVES

- To understand the fundamentals and architecture of Internet of Things
- To learn about the sensors and different layer protocols
- To learn the best practices in security and data analytics in IoT infrastructure
- To study the concept of Internet of Things in the real-world applications.

UNIT I INTRODUCTION TO IoT 9

Genesis -- Impact and Challenges of IoT -- IoT Network Architecture and Design: Need for new architectures -- Basic IoT Architecture -- foneM2M and IoT world forum architectures -- Core IoT functional stack -- IoT data management and compute stack.

UNIT II ENGINEERING IoT NETWORKS 9

Sensing Devices: Sensors -- Actuators -- MEMS -- Smart objects -- Sensor networks -- Connecting smart objects; IoT Access Technologies: IEEE 802.15.4 -- IEEE 802.15.4g and 802.15.4e -- IEEE 1901.2a -- IEEE 802.11ah -- LoRaWAN -- NB-IoT and other LTE variations.

UNIT III IoT NETWORK AND APPLICATION LAYER PROTOCOLS 9

IP as IoT network layer -- Adoption or adaption of IP -- Need for optimization -- Optimizing IP for IoT -- Profiles and Compliances; IoT Application Transport Methods: Non-Application Layer -- SCADA -- Web Based Protocols -- IoT Application Layer Protocols: CoAP and MQTT.

UNIT IV DATA ANALYTICS FOR IoT 9

Introduction -- Machine Learning -- Big Data Analytics Tools and Technologies -- Edge Streaming Analytics -- Network Analytics.

UNIT V IoT SECURITY ANDCASE STUDY 9

Cyber Security Vernacular -- Anatomy of IoT Cyber Attacks -- Physical and Hardware Security -- IoT Security and Best Practices; Case Study: Smart and connected cities.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Explain the basics and different architectures of IoT (K2)
- Discuss various sensors and access protocols for IoT (K2)
- Select different transport and application layer protocols based on application's requirements (K2)
- Discuss various machine learning and data analytics techniques for IoT applications (K2)
- Use security aspects in designing real time IoT applications (K3).

TEXTBOOKS

1. David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Robert Barton, Jerome Henry, “IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things”, Cisco Press, Pearson, 2019.
2. Perry Lea, “Internet of Thingsfor Architects”, Packt Publishing, O’Reilly, January 2018.

REFERENCE BOOKS

1. ArshdeepBahga,Vijay Madisetti, “Internet of Things:A hands-on Approach”, Universities Press, 2015.
2. Dieter Uckelmann, Mark Harrison, Michahelles, Florian (Eds), “Architecting the Internet of Things”, Springer, 2011.
3. HonboZhou, “The Internet of Things in the Cloud: A Middleware Perspective”, CRC Press, 2012.
4. Jan Holler,Vlasios Tsiatsis, Catherine Mulligan, Stamatis, Karnouskos, Stefan Avesand, DavidBoyle, “From Machine-to-Machine to the Internet ofThings – Introduction to a New Age of Intelligence”, Elsevier, 2014.
5. Olivier Hersent, David Boswarthick, Omar Elloumi, “The Internet of Things– Key applications and Protocols”, Wiley, 2012.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1625	FOUNDATIONS OF DATASCIENCE	3	0	0	3

OBJECTIVES

- To learn fundamentals of Data Science using Python
- To understand probability distributions and statistical Inferences
- To be familiar with supervised and unsupervised methods in machine learning
- To explore the algorithms used for analysing massive data problems and social networks
- To learn about topic and graphical models.

UNIT I DATA SCIENCE ANDPYTHON 9

Introduction: Computational tools -- Need for data science -- Causality and experiments; Array Computing in Python: Vectors -- Arrays -- Advanced vectorization of functions -- Higher-dimensional Arrays: Matrices and arrays; Dictionaries and Strings.

UNIT II PROBABILITY ANDSTATISTICS 9

Randomness -- Empirical Distributions -- Testing Hypothesis -- Estimation -- Why the mean matters -- Prediction -- Inference for Regression.

UNIT III MACHINE LEARNING 9

Perceptron algorithm -- Kernel functions -- Overfitting and uniform convergence -- Regularization -- Support Vector Machines -- Strong and weak learning -- Stochastic Gradient Descent.

UNIT IV DATA STREAMS ANDCLUSTERING 9

Algorithms for Massive Data Problems: Frequency moments of data streams -- Matrix algorithms using sampling; Clustering: k-Means clustering -- Spectral clustering -- Community finding and graph partitioning.

UNIT V TOPIC MODELS ANDGRAPHICAL MODELS 9

Topic Models -- Nonnegative matrix factorization -- Latent Dirichlet allocation -- Hidden Markov models -- Bayesian Belief Networks -- Markov Random Fields.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Develop Python programs to perform analysis on data (K3)
- Understand various probability distributions and statistical inferences (K2)
- Develop applications to demonstrate machine learning algorithms in practice (K3)
- Understand the principles of handling data streams (K2)
- Discuss topic and graphical modeling techniques in real world problem (K2).

TEXTBOOKS

1. AniAdhikari, JohnDeNero, “Computational and Inferential Thinking: The Foundations of Data Science”, GitBook, 2017. (Unit- I, II)

2. AvrimBlum,JohnHopcroft, Ravindran Kannan, “Foundations of Data Science”, VorabversioneinesLehrbuchs, 2016. (Unit-III, IV, V)

REFERENCE BOOKS

1. Hans Petter Langtangen, “A Primer on Scientific Programming with Python”, 4th Edition, Springer, 2016. (Unit - I).
2. Jonathan Dinu,“Foundations of DataScience:A PracticalIntroduction to DataScience withPython”, Addison-Wesley Data& Analytics Series,2016.
3. JureLeskovek, Anand Rajaraman, Jeffrey Ullman, “Mining of Massive Datasets”, V2.1, Cambridge University Press, 2014.
4. EMC Education Services, “Data Science and Big Data Analytics:Discovering, Analyzing, Visualizing and Presenting Data”, Wiley publishers, 2015.
5. Cathy O’Neil, Rachel Schutt.“Doing Data Science, Straight Talk from TheFrontline”, O’Reilly, 2014.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1626	CLOUD COMPUTING	3	0	0	3

OBJECTIVES

- To understand the concepts of virtualization and virtual machines
- To learn about virtualization infrastructure
- To understand the principles of Cloud Architecture, Models and Infrastructure
- To explore and experiment various Cloud deployment environments
- To learn about the security issues in the cloud environment.

UNIT I CLOUDARCHITECTUREMODELSANDINFRASTRUCURE 8

Cloud Architecture: System Models for Distributed and Cloud Computing -- NIST Cloud Computing Reference Architecture -- Cloud deployment models -- Cloud service models; Cloud Infrastructure: Architectural design of compute and storage clouds -- Layered cloud architecture Development -- Design Challenges -- Inter Cloud Resource Management -- Resource Provisioning and Platform Deployment.

UNIT II INTRODUCTION TO VIRTUALIZATION 9

Basics of Virtualization -- Emulation -- Interpretation -- Binary Translation -- Virtualization Advantages -- Virtual Machine Basics -- Taxonomy of Virtual Machines -- Process Virtual Machines -- System Virtual Machines -- Hypervisor -- Key Concepts -- Virtualization structure -- Implementation levels of virtualization -- Virtualization Types: Full Virtualization -- Para Virtualization -- Hardware Virtualization.

UNIT III VIRTUALIZATIONINFRASTRUCTURE 9

Comprehensive Analysis -- Resource Pool -- Testing Environment -- Virtual Workloads -- Provision of Virtual Machines -- Desktop Virtualization -- Network Virtualization -- Server and Machine Virtualization -- Storage Virtualization -- System-level of Operating Virtualization -- Application Virtualization-- Virtualization of CPU, Memory and I/O devices -- Virtual clusters and Resource Management -- Virtual Machine Monitors: KVM, Xen, VMWareESXi server.

UNIT IV CLOUD DEPLOYMENT ENVIRONMENT 12

Parallel Programming Framework: Hadoop Map Reduce -- Google App Engine -- Amazon AWS -- Microsoft Azure; Cloud Software Environments -- Eucalyptus -- OpenStack -- OpenNebula -- Aneka -- CloudSim.

UNIT V CLOUD SECURITY 7

Data Security and Storage; Identity and Access Management(IAM) -- IAM Challenges -- IAM Architecture and Practice; Security Management in the Cloud -- Security Management Standards -- SaaS, PaaS and IaaS Availability Management -- Access Control; Security-As-A-[Cloud] Service.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Describe the design challenges in cloud environment (K2)
- Apply the concept of virtualization and analyse its types (K3)
- Experiment with virtualization of hardware resources and Virtual Machine Monitors (K3)
- Develop and deploy services on cloud and be able to set up a private cloud environment using open source software (K3)
- Understand security challenges in cloud environment (K2).

TEXTBOOKS

1. Kai Hwang, GeofferyC Fox, Jack J Dongarra, "Distributed and Cloud Computing: Clusters, Grids,Clouds and the Future of Internet", 1st Edition, Morgan Kaufman Publisher, an Imprint of Elsevier,2012. (Unit I, II, IV)
2. Srinivasan A, SureshJ, "Cloud Computing: A practical Approach for Learning andImplementation", Pearson Education India,2014. (Units I, II, III, IV)

REFERENCE BOOKS

1. Tim Mather, Subra Kumaraswamy, Shahed Latif, "Cloud Security and Privacy: an enterprise perspective on risks and compliance", O'Reilly Media, 2009. (Unit V)
2. Danielle Ruest, Nelson Ruest, "Virtualization: A Beginner's Guide", McGraw- Hill Osborne Media,2009.
3. James E Smith, Ravi Nair, "Virtual Machines: Versatile Platforms for Systems and Processes", Elsevier/Morgan Kaufmann, 2005.
4. William von Hagen, "Professional Xen Virtualization", Wrox Publications, January 2008.
5. David Marshall, Wade A Reynolds, "Advanced Server Virtualization: VMware and Microsoft Platform in the Virtual Data Center", Auerbach Publications, 2006.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1721	SERVICE ORIENTED ARCHITECTURE	3	0	0	3

OBJECTIVES

- To learn the fundamentals of XML
- To understand Service Oriented Architecture, Web services and their importance
- To know the web services standards and technologies
- To study the web service extensions
- To develop SOA based applications using service-oriented analysis and design.

UNIT I XML TECHNOLOGIES 9

XML Document Structure: Well-formed and valid documents -- DTD -- XML Schema; Parsing XML using DOM -- SAX; XPath -- XML transformation and XSLT -- Xquery.

UNIT II SERVICE ORIENTED ARCHITECTURE BASICS 9

Characteristics of SOA -- Benefits of SOA -- Comparing SOA with client server and distributed architectures -- Principles of service orientation -- Service layers.

UNIT III WEB SERVICES ANDSTANDARDS 8

Web Services Platform -- Service descriptions -- WSDL -- Messaging with SOAP -- Service discovery -- UDDI -- Service level interaction patterns -- Orchestration and choreography.

UNIT IV WEB SERVICES EXTENSIONS 8

WS-Addressing -- WS-Reliable messaging -- WS-Policy -- WS-Coordination -- WS-Transactions -- WS-Security -- Examples -- XML web services for .Net.

UNIT V SERVICE ORIENTED ANALYSIS ANDDESIGN 11

Service oriented enterprise applications -- Service oriented analysis and design: Need for models -- Principles of service design -- Design of activity services -- Design of data services -- Design of client services -- Design of business process services.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Understand XML technologies (K2)
- Understand service orientation, benefits of SOA (K2)
- Understand web services and WS standards (K2)
- Apply web services extensions to develop solutions(K3)
- Understand and apply service modeling, service-oriented analysis and design for application development (K3).

TEXTBOOKS

1. ThomasErl, “Service Oriented Architecture: Concepts, Technology, andDesign”, Pearson Education, 2005.

1. 2.SahankarKambhampaly,“Service-Oriented Architecture for Enterprise Applications”, Wiley India Pvt.Ltd,2008

REFERENCE BOOKS

1. James McGovern, Sameer Tyagi, Michael E Stevens, Sunil Mathew, “Java Web Services Architecture”, Elsevier, 2003.
2. Ron Schmelzer et al, “XML and Web Services”, Pearson Education, 2002.
3. FrankP Coyle, “XML, Web Services and the Data Revolution”, Pearson Education, 2002.
4. Sandeep Chatterjee, James Webber, “Developing Enterprise Web Services: An Architect’sGuide”, Prentice Hall, 2004.
5. Eric Newcomer, Greg Lomow, “Understanding SOA with Web Services”, Pearson Education, 2005.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1722	SOCIAL NETWORK ANALYSIS	3	0	0	3

OBJECTIVES

- To understand the concepts of social network analysis and formally represent social network
- To use SNA tools for applying community detection algorithms and visualization on online social network
- To know the various applications of social network analysis.

UNIT I INTRODUCTION 9

Social Network Analysis: Development of social network analysis -- Key concepts and measures in network analysis -- Electronic Sources for Network Analysis: Electronic discussion networks -- Blogs and online communities -- Web-based networks -- Social Network Data: Introduction -- Boundary specification and sampling -- Types of networks -- Network data -- Measurement and collection.

UNIT II MATHEMATICAL REPRESENTATION OF SOCIAL NETWORKS 9

Notations for Social Networks: Graph theoretic notations -- Sociometric notations -- Algebraic notations -- Two sets of actors -- Graph and matrices.

UNIT III COMMUNITY DETECTION METHODOLOGIES, APPLICATIONS 9

Introduction -- Definition of communities -- Evaluating communities -- Methodologies of Network Community Mining: Optimization based algorithms -- Heuristic methods -- Other methods -- Applications of community mining algorithms -- Multi-Relational characterization of dynamic social network communities.

UNIT IV PRACTICAL APPROACH TO SOCIAL NETWORK ANALYSIS 9

Graph Theory: Introduction; SNA Tool: Python and NetworkX -- Centrality; Clique, Clusters and Components: Components and Subgraphs -- Triads -- Cliques -- Hierarchical Clustering; 2-Mode networks; A dynamic model in Python.

UNIT V VISUALIZATION AND APPLICATIONS OF SOCIAL NETOWRKS 9

Visualizing online social networks -- Visualizing social networks with matrix-based representations -- Node-Edge diagrams -- Matrix and Node-Link Diagrams -- Hybrid representations; Applications: Covert networks -- Community welfare -- Collaboration networks -- Co-citation networks.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Explain Social network concepts, measures and data (K2)
- Represent social networks mathematically (K2)

- Understand the working of community detection algorithms for online Social networks (K2)
- Use SNA tools for analysing social networks (K3)
- Apply Visualization to social networks (K3)

TEXTBOOKS

1. Peter Mika, "Social Networks and the Semantic Web", 1st Edition, Springer,2007.
1. (Unit I)
2. StanleyWasserman, Katherine Faust, "Social Network Analysis Methods and Applications", 1st Edition,Cambridge University Press, 1994. (Unit I and Unit II)
3. Maksim Tsvetovat and Alexander Kouznetsov, "Social Network Analysis for Startups", O'Reilly, 2011. (Unit IV)
4. BorkoFurht, "Handbook of Social Network Technologies andApplications",1st Edition,Springer, 2010. (Unit III and Unit V)

REFERENCE BOOKS

1. John Scott, "Social Network Analysis", 4th Edition,SAGE Publications,2017.
2. Stephen P Borgatti, MartinG Everett,Jeffrey G Johnson, "Analyzing Social Networks", SAGE Publications, 2nd Edition,2018.
3. Robert A Hanneman, Mark Riddle, "Introduction to social network methods", University of California, Riverside, 2005.
4. Charles Kadushin, "Understanding Social Networks: Theories, Concepts, and Findings", 1st Edition,Kindle Edition,Oxford University Press, 2012.
5. Guandong Xu, Yanchun Zhang, Lin Li, "Web Mining and Social Networking – Techniques and applications", 1st EditionSpringer, 2011.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1723	DEEP LEARNING	3	0	0	3

OBJECTIVES

- To understand the basics of deep neural networks
- To understand CNN and RNN architectures of deep neural networks
- To comprehend advanced deep learning models
- To learn the evaluation metrics for deep learning models.

UNIT I DEEP NETWORKS BASICS 9

Linear Algebra: Scalars -- Vectors -- Matrices and tensors; Probability Distributions -- Gradient-based Optimization -- Machine Learning Basics: Capacity -- Overfitting and underfitting -- Hyperparameters and validation sets -- Estimators -- Bias and variance -- Stochastic gradient descent -- Challenges motivating deep learning; Deep Networks: Deep feedforward networks; Regularization -- Optimization.

UNIT II CONVOLUTIONAL NEURAL NETWORKS 9

Convolution Operation -- Sparse Interactions -- Parameter Sharing -- Equivariance -- Pooling -- Convolution Variants: Strided -- Tiled -- Transposed and dilated convolutions; CNN Learning: Nonlinearity Functions -- Loss Functions -- Regularization -- Optimizers -- Gradient Computation.

UNIT III RECURRENT NEURAL NETWORKS 10

Unfolding Graphs -- RNN Design Patterns: Acceptor -- Encoder -- Transducer; Gradient Computation -- Sequence Modeling Conditioned on Contexts -- Bidirectional RNN -- Sequence to Sequence RNN -- Deep Recurrent Networks -- Recursive Neural Networks -- Long Term Dependencies; Leaky Units: Skip connections and dropouts; Gated Architecture: LSTM.

UNIT IV MODEL EVALUATION 8

Performance metrics -- Baseline Models -- Hyperparameters: Manual Hyperparameter -- Automatic Hyperparameter -- Grid search -- Random search -- Debugging strategies.

UNIT V AUTO ENCODERS AND GENERATIVE MODELS 9

Autoencoders: Undercomplete autoencoders -- Regularized autoencoders -- Stochastic encoders and decoders -- Learning with autoencoders; Deep Generative Models: Variational autoencoders -- Generative adversarial networks.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Understand basics in deep neural networks (K2)
- Apply Convolution Neural Network for image processing (K3)
- Apply Recurrent Neural Network and its variants for text analysis(K3)
- Apply model evaluation for various applications (K3)
- Understand the concepts in autoencoders and generative models (K2).

TEXTBOOKS

1. Ian Goodfellow, Yoshua Bengio, Aaron Courville, “Deep Learning”, MIT Press, 2016.

REFERENCE BOOKS

1. SalmanKhan, Hossein Rahmani, Syed Afaq Ali Shah, Mohammed Bennamoun, “A Guideto Convolutional Neural Networks for Computer Vision”,Synthesis Lectures on Computer Vision, Morgan & Claypool publishers,2018.
2. Yoav Goldberg, “Neural Network Methods for Natural Language Processing”,Synthesis Lectures on Human Language Technologies, Morgan& Claypool publishers, 2017.
3. Francois Chollet, “Deep Learning with Python”, Manning Publications Co, 2018.
4. Charu C. Aggarwal, “Neural Networks and Deep Learning: A Textbook”, Springer International Publishing, 2018.
5. Josh Patterson, Adam Gibson, “Deep Learning: A Practitioner ’s Approach”, O’Reilly Media,2017.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1724	MULTICORE ARCHITECTURE AND PROGRAMMING	3	0	0	3

OBJECTIVES

- To understand the need for multi-core processors, and their architecture
- To understand the challenges in parallel and multi-threaded programming
- To learn about the various parallel programming paradigms
- To develop OpenMP programs and design parallel solutions
- To develop an application using MPI programming.

UNIT I MULTI-CORE PROCESSORS 9

Single core to Multi-Core Architectures; SIMD and MIMD Systems; Interconnection Networks; Symmetric and Distributed Shared Memory Architectures -- Cache Coherence -- Performance Issues – Parallel Program Design.

UNIT II PARALLEL PROGRAMMING 9

Performance -- Scalability; Synchronization and Data Sharing -- Data Races -- Synchronization Primitives (mutexes, locks, semaphores, barriers); Deadlocks and Livelocks; Communication Between Threads (condition variables, signals, message queues and pipes).

UNIT III SHARED MEMORY PROGRAMMING WITH OPENMP 9

OpenMP Execution Model: Memory model -- OpenMP Directives -- Work-Sharing Constructs -- Library Functions -- Handling Data and Functional Parallelism -- Handling Loops – Performance Considerations.

UNIT IV DISTRIBUTED MEMORY PROGRAMMING WITH MPI 9

MPI program execution: MPI constructs -- Libraries; MPI Send and Receive -- Point-to-point and collective communication; MPI derived datatypes -- Performance evaluation.

UNIT V PARALLEL PROGRAM DEVELOPMENT 9

Case studies: n-Body solvers; Tree Search -- OpenMP and MPI implementations and comparison.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Understand the limitations of single core processors and the concepts behind the various multi-core architectures (K2)
- Identify the issues in programming Parallel Processors (K2)
- Develop the programs using OpenMP (K3)
- Develop the program using MPI (K3)
- Compare and contrast programming for serial processors and programming for parallel processors (K3).

TEXTBOOKS

1. PeterS Pacheco, “An Introduction to Parallel Programming”, Morgan Kaufmann/Elsevier, 2011.
2. DarrylGove, “Multicore Application Programming for Windows, Linux, and Oracle Solaris”, Pearson, 2011.

REFERENCE BOOKS

1. MichaelJ Quinn, “Parallel programming in C withMPI andOpenMP”, Tata McGraw-Hill, 2003.
2. Victor Alessandrini, “Shared Memory Application Programming,Concepts and Strategies in Multicore Application Programming”, 1st Edition, Morgan Kaufmann, 2015.
3. Yan Solihin, “Fundamentals of Parallel Multicore Architecture”, CRC Press,2015.
4. RohitChandra, Ramesh Menon, Leo Dagum, David Kohr, Dror Maydan and Jeff McDonald, “Parallel Programming in OpenMP”, 1st Edition, Morgan Kaufmann, 2000.
5. GerassimosBarlas, “Multicore and GPU Programming”, Morgan Kaufmann,2014.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1725	LOGIC IN COMPUTER SCIENCE	3	0	0	3

OBJECTIVES

- To understand the role of logic in modeling and reasoning about systems
- To learn the syntax and semantics of various logics
- To write formal specifications in various logics
- To learn resolution for propositional and predicate logic
- To learn natural deduction for modal logic
- To program using SAT-SMT Solvers.

UNIT I PROPOSITIONAL LOGIC 9

Foundations: Syntax and semantics -- Mathematical induction -- Soundness and completeness; Semantic Entailment: Natural Deduction; Normal Forms; PSAT Problem; Horn Logic; DPLL; CDCL.

UNIT II PREDICATE LOGIC 9

Foundations: Syntax and semantics; Semantic Entailment: Natural deduction; Normal Forms; Undecidability of Predicate Logic; Herbrand's Theory.

UNIT III MODAL LOGIC 9

Modes of truth; Basic Modal Logic: Syntax -- Semantics; Logic Engineering: The stock of valid formulas -- Important properties of the accessibility relation -- Correspondence theory -- Some Modal Logics; Natural deduction; Reasoning about Knowledge in a Multi-agent System:Examples -- Modal Logic KT45n -- Natural deduction for KT45n -- Formalizing the examples.

UNIT IV SAT SMT SOLVERS 9

Resolution: Ground resolution -- Unification; Satisfiability Solvers: Definitions and Notations -- SAT solver technology -- Complete methods; Satisfiability Modulo Theories: Introduction -- Background -- Eager encoding to SAT -- Integrating theory solvers into SAT engines.

UNIT V MODEL CHECKING 9

Motivation for verification; Linear-time Temporal Logic: Syntax -- Semantics -- Practical patterns of specifications; Model Checking: The NuSMV model checker -- Case studies; Branching-time Logic: Syntax -- Semantics -- Practical patterns of specifications; Model-checking Algorithms: The CTL model-checking algorithm -- The LTL model-checking algorithm; CTL* and the expressive powers of LTL and CTL.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Write specifications in predicate logic and temporal logics (K3)
- Apply resolution to solve semantic entailment (K3)
- Write programs using SAT-SMT Solvers (K3)
- Specify and Verify simple systems using NuSMV(K3)

- Use Natural Deduction for modal logic theories(K3).

TEXTBOOKS

1. M. Huth, M Ryan, "Logic in Computer Science–Modeling and Reasoning aboutsystems", 2nd Edition, Cambridge University Press,2004 .(Units I,II,III and V).
2. Uwe Schoning "Logic for Computer Scientists", Birkhauser, 1989 (UnitsI,II,IV).
3. CarlaGomes, Henry Kautz, Ashish Sabharwal and Bart Selman "Satisfiability Solvers", Handbook of Knowledge Representation, Elsevier2008 (Unit IV).

REFERENCE BOOKS

1. ClarkBarrett, Roberto Sebastiani, Sanjit Seshia, Cesare Tinelli, "Satisfiability Modulo Theories", Handbook of Satisfiability, IOS Press 2009 (Unit IV).
2. M. Ben-Ari, "Mathematical logic for computer science", 2nd Edition, Springer,2003.
3. Arindama Singh, "Logics for Computer Science", 2nd edition, PHI, 2018.
4. StanleyN Burris, "Logic for Mathematics & Computer Science", Prentice Hall 1998.
5. Herbert Enderton, "A Mathematical Introduction to Logic", 2nd edition, Harcourt Academic Press, 2001.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1726	SOFTWARE ARCHITECTURE	3	0	0	3

OBJECTIVES

- To understand the various architectural views and quality attributes.
- To learn the architectural styles
- To study the architectural description languages
- To learn the architecture evaluation techniques
- To relate software architecture and software quality attributes.

UNIT I ARCHITECTURAL VIEWS AND QUALITY ATTRIBUTES 9

Basic Concepts of Software Architecture – Architecture Business Cycle – Architectural Patterns – Reference Models – Architectural Structures, Views; Module Views, Component-Connector Views, Allocation Views – Understanding Quality Attributes – Functionality and Architecture – Architecture and Quality Attributes – System Quality Attributes – Quality Attribute Scenarios in Practice.

UNIT II ARCHITECTURAL STYLES 9

Introduction to Architectural Styles – Simple Styles – Distributed and Networked Architectures – Architecture for Network based Applications – Decentralized Architectures.

UNIT III ARCHITECTURE DESCRIPTION, DOCUMENTATION AND EVALUATION 9

Early Architecture Description Languages – Domain and Style Specific ADLs – Extensible ADLs – Documenting Software Architectures – Architecture Evaluation – ATAM.

UNIT IV CREATING ARCHITECTURE 9

Introducing Tactics – Availability Tactics – Modifiability Tactics – Performance Tactics – Security Tactics – Testability Tactics – Usability Tactics – Relationship of Tactics to Architectural Patterns – Architectural Patterns and Styles.

UNIT V CASE STUDIES 9

Distributed Architectures for Data-intensive Systems that use Micro services – Architecture of Software Systems involving Internet-of-Things (IoT).

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Describe various architectural views and quality attributes (K2)
- Apply suitable architectural styles based on the software quality requirements (K3)
- Define ADLs and evaluate alternate architectural solutions (K4)
- Construct new architectures based on the quality attribute requirements (K4)
- Analyze the architectural decisions for building data intensive and IoT systems(K4)

TEXTBOOKS

1. Len Bass, Paul Clements, Rick Kazman, “Software Architecture in Practice”, Third Edition, Addison Wesley, 2012.
2. Richard N Taylor, Nenad Medvidovic, Eric M Dashofy, “Software Architecture: Foundations, Theory and Practice”, Wiley 2010.

REFERENCE BOOKS

1. Mary Shaw, David Garlan, “Software Architecture: Perspectives on an Emerging Discipline”, Pearson Education, 2008.
2. Kai Qian et al., “Software Architecture and Design Illuminated”, Jones and Bartlett Publishers, Canada, 2010
3. Martin Kleppman, “Designing Data-Intensive Applications”, O’Reilly Media, 2017.
4. https://www.researchgate.net/publication/260542337_IoT_Reference_Architecture.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1727	GPU COMPUTING	3	0	0	3

OBJECTIVES

- To understand the basics of GPU architectures
- To write programs for massively parallel processors
- To understand the issues in mapping algorithms for GPUs
- To introduce different GPU programming models
- To understand the concepts of OpenCL

UNIT I GPU ARCHITECTURE 12

Evolution of GPU architectures -- Understanding Parallelism with GPU -- Typical GPU Architecture -- CUDA Hardware Overview -- Threads, Blocks, Grids, Warps, Scheduling -- Memory Handling with CUDA: Shared memory, Global memory, Constant memory and Texture memory.

UNIT II CUDA PROGRAMMING 8

Using CUDA -- Multi GPU -- Multi GPU Solutions -- Optimizing CUDA Applications: Problem decomposition, Memory considerations, Transfers, Thread usage, Resource contentions.

UNIT III PROGRAMMING ISSUES 8

Common Problems: CUDA error handling, Parallel programming issues, Synchronization, Algorithmic issues, Finding and avoiding errors.

UNIT IV OPENCL BASICS 8

OpenCL Standard -- Kernels -- Host Device Interaction -- Execution Environment -- Memory Model -- Basic OpenCL Examples.

UNIT V ALGORITHMS ON GPU 9

Parallel Patterns: Convolution, Prefix Sum, Sparse Matrix -- Matrix Multiplication -- Programming Heterogeneous Cluster.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Understand GPU architecture (K2)
- Write programs using CUDA, identify issues and debug them (K3)
- Implement efficient algorithms in GPUs for common application kernels such as matrix multiplication (K3)
- Write simple programs using OpenCL (K3)
- Write an efficient parallel program for a given problem(K3).

TEXTBOOKS

1. Shane Cook, “CUDA Programming: A Developer ’s Guide to Parallel Computing withGPUs (Applications of GPU Computing)”, 1st Edition,Morgan Kaufmann, 2012.
2. DavidRKaeli, Perhaad Mistry, Dana Schaa, Dong Ping Zhang, “Heterogeneous Computing with OpenCL”, 3rd Edition, Morgan Kauffman, 2015.

REFERENCE BOOKS

1. DavidB Kirk, Wen-mei W Hwu, “Programming Massively parallel Processors – A Hands-on Approach”, 3rd Edition,Morgan Kaufmann, 2016.
2. Nicholas Wilt, “CUDA Handbook: A Comprehensive Guideto GPU Programming”, Addison - Wesley, 2013.
3. Jason Sanders, Edward Kandrot, “CUDA by Example: An Introduction to General Purpose GPU Programming”, Addison - Wesley, 2010.
4. http://www.nvidia.com/object/cuda_home_new.html
5. http://www.openCL.org

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1728	USER EXPERIENCE DESIGN	3	0	0	3

OBJECTIVES

- To develop skills in analyzing the UX in agile development
- To impart the skills required to create an Information Architecture document for a agile development
- To establish requirements for User Experience design concepts using techniques such as user stories
- To learn the agile concepts used by UX team in terms of backlog, feedback, and communication.

.

UNIT I CONCEPTUALIZING USER EXPERIENCE DESIGN 8

What are UX and UX design: Definition of UX -- UX design -- components of UX -- What UX is Not -- Kinds of interaction and UX; UX processes, Lifecycles, Methods and Techniques: Basic process components for UX -- fundamental UX lifecycle activities -- UX design techniques as life skills -- Choosing UX processes, methods and techniques; Agile lifecycle processes and the funnel model of Agile UX: Embracing an agile lifecycle process -- funnel model of Agile UX -- Agile UX case study.

UNIT II DATA MODELING AND PROTOTYPING 9

Data Modeling: User work role model -- Flow model -- Task structure models -- Artifact model -- Physical work environment model -- Information architecture model -- social model-- Hybrid models -- Model consolidation; UX design requirements: User stories -- UX Design requirements -- validating user stories and requirements; Prototype candidate design: Depth and Breadth of a prototype -- fidelity -- wireframe -- specialized prototypes -- software tools.

UNIT III UX DESIGN 9

Nature of UX Design: What is Design -- Design lifecycle for the agile UX funnel -- Bottom up Design -- Top up design -- Generative design; Mental models and conceptual design: Conceptual Design works as a connection of mental models; Designing the interaction: Creating an interaction design -- storyboards-- wireframes--intermediate interaction design -- interaction design production -- case study.

UNIT IV UX EVALUATION METHODS 8

UX Evaluation methods: Data Collection techniques -- UX evaluation methods; Data analysis: analyze Quantitative data -- analyze qualitative UX data -- Reporting different kinds of data; Agile concepts for UX teams: creating a user experience backlog -- constant feedback and iteration -- thinking and communicating in terms of user stories -- defining acceptance criteria.

UNIT V ITERATION AND CASE STUDIES 10

Iterations: working as a team -- design documentation -- working with the product owner -- working in iterations -- continuous improvement; Toolbox: As-is experience design review -- as-is/to-be process mapping -- camera as documentation -- collaborative design -- competitor

review -- context scenarios -- customer experience -- customer testing -- task analysis -- trade-off sliders -- case study.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Identify the users and learn the entire user experience lifecycle of agile UX design (K2)
- Develop a deep understanding of UX design and evaluation (K2)
- Create efficient prototype to communicate and evaluate the design definition (K3)
- Apply UX design in a case study (K3)
- Learn the customer experience and testing (K4)

TEXTBOOKS

1. Rex Hartson, Pardha Pyla, "The UX book: Agile UX design for a Quality User Experience", Morgan Kaufmann Publishers, Elsevier,2nd Edition, 2019 (Unit I, II, III, IV).
2. Lindsay Ratcliffe and Marc McNeill, "Agile Experience Design: A Digital Designers Guide to Agile, Lean and Continuous", Newriders, Berkeley, CA, 2012 (Unit V).

REFERENCE BOOKS

1. Diana De Marco Brown, "Agile User Experience Design: A Practitioner's Guide to Makingit Work",Morgan Kaufmann, Elsevier, USA, 2013 (Unit IV).
2. Pieter Jongerlus and Annaoffermans, "Get Agile! Scrum for UX, Design and Development", BIS publishers, Amsterdam, Netherlands, 2012.
3. Jeffy Gothelf, Josh Seiden, "Lean UX Designing Great Products with Agile Teams", Second edition, O'Reilly Media Inc, CA, 2016.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1729	DATA WAREHOUSING AND DATA MINING	3	0	0	3

OBJECTIVES

- To understand data warehouse concepts, architecture, business analysis and tools
- To understand data pre-processing and data visualization techniques
- To study algorithms for finding hidden and interesting patterns in data
- To understand and apply various classification techniques
- To understand clustering techniques.

UNIT I DATA WAREHOUSING ANDBUSINESS ANALYSIS 9

Basic Concepts: Data warehousing components -- Building a data Warehouse -- Database Architectures for Parallel Processing -- Parallel DBMS Vendors;Multidimensional Data Model; Data Warehouse Schema for Decision Support;Concept Hierarchies; Characteristics of OLAP Systems -- Typical OLAP Operations, OLAP and OLTP.

UNIT II INTRODUCTION TO DATA MINING 9

Introduction to Data Mining Systems -- Knowledge Discovery Process -- Data Mining Techniques -- Issues -- applications; Data Objects and attribute types -- Statistical descriptionof data;Data Preprocessing: Cleaning -- Integration -- Reduction -- Transformation -- Discretization; Data Visualization; Data similarity and dissimilarity measures.

UNIT III FREQUENT PATTERN ANALYSIS 9

Mining Frequent Patterns -- Associations and Correlations -- Mining Methods -- Pattern Evaluation Method; Pattern Mining in Multilevel -- Multi-Dimensional Space -- Constraint Based Frequent Pattern Mining; Classification using Frequent Patterns.

UNIT IV CLASSIFICATION 9

Introduction -- Decision Tree Induction; Bayesian Classification; Support Vector Machines; Lazy Learners; Model Evaluation and Selection -- Techniques to improve Classification Accuracy; Evaluation of Classification Methods; Datasets.

UNIT V CLUSTERING 9

Introduction -- Clustering -- Cluster Analysis -- Clustering Methods: K-means-- Hierarchical clustering -- Agglomerative clustering -- Evaluating clusters; Case Studies using WEKA tools for classification -- Clustering and association.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Design a Data warehouse system and perform business analysis with OLAP tools (K3)
- Apply suitable pre-processing techniques (K3)
- Apply frequent pattern and association rule mining techniques for data analysis (K3)
- Apply appropriate classification techniques for data analysis (K3)

- Apply clustering techniques using appropriate tools (K3).

TEXTBOOKS

1. Jiawei Han, Micheline Kamber, "Data Mining Concepts and Techniques", 3rdEdition,Elsevier, 2012.
2. Alex Berson, Stephen J Smith, "Data Warehousing, Data Mining & OLAP", Tata McGraw-Hill Edition,35th Reprint 2016.

REFERENCE BOOKS

1. Pang-Ning Tan, MichaelSteinbach,VipinKumar, "Introduction to Data Mining", PersonEducation, 2007.
2. K P Soman, Shyam Diwakar, V Ajay, "Insight into Datamining Theory and Practice",Eastern Economy Edition, Prentice Hall of India,2006.
3. G K Gupta, "Introduction to Data Mining with Case Studies", Easter Economy Edition,PrenticeHall of India,2006.
4. Daniel T Larose, "Data Mining Methods and Models", Wiley-Inderscience, 2006.
5. Ian H Witten, Eibe Frank, "Data Mining: Practical Machine Learning Tools and Techniques", Elsevier, 2nd Edition,2016.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1731	SOFT COMPUTING	3	0	0	3

OBJECTIVES

- To learn the basic concepts of Soft Computing
- To study various Artificial Neural network architectures
- To learn fuzzy sets, fuzzy logic and fuzzy inference system
- To understand genetic algorithm for global optimization
- To learn hybrids of neuro, fuzzy and genetic algorithm, and their applications

UNIT I INTRODUCTION TO SOFT COMPUTING 8

Introduction: Neural networks -- Fuzzy logic -- Genetic algorithm -- Hybrid systems; Artificial Neural Network: Fundamental concepts -- Evolution of neural networks -- Basic modals of ANN -- McCulloch and Pitts neuron -- Linear separability -- Hebb network.

UNIT II SUPERVISED,UNSUPERVISEDANDASSOCIATIVELEARNING NETWORKS 11

Supervised Learning Network: Perceptron networks -- Adaptive linear neuron -- Multiple adaptive linear neurons -- Back propagation networks -- Radial bias function network; Associative Memory Networks: Auto associative memory network -- Bidirectional associative memory -- Hopfield networks; Unsupervised Learning Networks: Hamming network -- Kohonen neural network -- Learning vector quantization -- Adaptive resonance theory networks.

UNIT III FUZZY SYSTEMS 9

Introduction to fuzzy logic -- Classical sets -- Fuzzy sets -- Fuzzy relations -- Membership functions -- Defuzzification methods -- Fuzzy arithmetic -- Fuzzy measures -- Fuzzy rule base and approximate reasoning -- Fuzzy decision making.

UNIT IV GENETIC ALGORITHMS 8

Genetic Algorithm and search space -- General genetic algorithm -- Operators -- Stopping condition -- Constraints -- Classification -- Genetic programming; Applications of genetic algorithm.

UNIT V HYBRID SOFT COMPUTINGTECHNIQUES & APPLICATIONS 9

Neuro-Fuzzy hybrid systems -- Genetic neuro hybrid systems -- Genetic fuzzy hybrid and fuzzy genetic hybrid systems; Applications of Soft Computing: A fusion approach of multispectral images with SAR -- Optimization of Traveling Salesman Problem using genetic algorithm -- Soft computing-based hybrid fuzzy controllers.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Understand various soft computing techniques (K2)
- Design and develop different neural network algorithms (K3)
- Analyse and apply fuzzy logic and fuzzy inference system (K3)
- Solve problems using Genetic Algorithms (K3)

- Apply various soft computing techniques for complex problems (K3).

TEXTBOOKS

1. S N Sivanandam, S N Deepa, "Principles of Soft Computing", Wiley India,2nd Edition, 2011.
2. Jyh-ShingRoger Jang, Chuen-Tsai Sun, Eiji Mizutani, "Neuro-Fuzzy and Soft Computing", Prentice-Hall of India, 2002.

REFERENCE BOOKS

1. Kwang H Lee, "First course on Fuzzy Theory and Applications", Springer,2005.
2. GeorgeJKlir, Bo Yuan, "Fuzzy Sets andFuzzyLogic-Theory andApplications", PrenticeHall, 1996.
3. James A Freeman, David M S kapura, "Neural Networks Algorithms, Applications, and Programming Techniques", Addison Wesley, 2003.
4. S Rajasekaran, G A Vijayalakshmi Pai, "Neural Networks, FuzzyLogic and Genetic Algorithm, Synthesis and Applications", PHI Learning, 2017.
5. N P Padhy, S P Simon, "Soft Computing with MATLAB Programming", OxfordUniversity Press, 2015.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1732	EMBEDDED SYSTEMS	3	0	0	3

OBJECTIVES

- To learn the architecture and programming of ARM processor
- To be familiar with the embedded computing platform design and analysis
- To study interfacing concepts
- To learn an embedded firmware and its designs
- To design embedded systems and to develop programs.

UNIT I EMBEDDED COMPUTINGANDARM PROCESSORS 9

Embedded Computing: Complex systems and microprocessors -- Embedded system design process -- Formalisms for system design -- Model train controller; Instruction Sets: Preliminaries -- ARM processor; CPUs: Programming input and output -- Supervisor mode, exceptions and traps -- Co-processors -- Memory system mechanisms -- CPU performance – CPU power consumption.

UNIT II EMBEDDED COMPUTINGPLATFORM DESIGN 9

Bus-Based Computer Systems: CPU Bus -- Memory devices and systems -- Designing with computing platforms -- Consumer electronics architecture -- Platform-level performance analysis; Program Design and Analysis: Components for embedded programs -- Models of programs -- Assembly, linking and loading -- Compilation techniques -- Program level performance analysis -- Software performance optimization -- Program level energy and power analysis and optimization -- Analysis and optimization of program size -- Program validation and testing.

UNIT III SENSOR INTERFACING WITH ARDUINO 9

Basics of hardware design and functions of basic passive components -- Sensors and Actuators -- Arduino code -- Library file for sensor interfacing -- Construction of basic applications.

UNIT IV EMBEDDED FIRMWARE 9

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

UNIT V EMBEDDED C PROGRAMMING 9

Introduction -- Reading switches -- Adding Structure to the code; Meeting Real-Time Constraints: Creating hardware delays using Timer 0 and Timer 1 -- Generating a Minimum and Maximum delay-Example -- Creating a portable hardware delay -- Timeout mechanisms -- Creating loop timeouts -- Testing loop timeouts -- Hardware timeouts -- Testing a hardware timeout.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Understand the architecture and programming of ARM processor (K2)

- Understand the concepts of embedded systems (K2)
- Understand peripherals and interfacing of sensors (K2)
- Apply the system design techniques to develop firmware (K3)
- Implement the code for constructing a system (K3).

TEXTBOOKS

1. Marilyn Wolf, "Computers as Components – Principles of Embedded Computing SystemDesign", 3rdEdition,Morgan Kaufmann Publisher (An imprint from Elsevier), 2012.
2. Michael J Pont, "Embedded C", 2nd Edition,Pearson Education, 2008.

REFERENCE BOOKS

1. Shibu K V, "Introduction to Embedded Systems", McGrawHill, 2014.
2. Jonathan W Valvano,"Embedded Microcomputer SystemsReal Time Interfacing", 3rd EditionCengage Learning, 2012.
3. Raj Kamal, "Embedded Systems-Architecture, Programming and Design",3rd edition, TMH, 2015.
4. Lyla, "Embedded Systems", Pearson, 2013.
5. J. M. Hughes, "Arduino: A TechnicalReference", O'Reilly Media,2016

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1733	GRAPHTHEORY AND APPLICATIONS	3	0	0	3

OBJECTIVES

- Be familiar with fundamental Graph Theory topics and results
- Be exposed to the techniques of proofs and analysis

UNIT I INTRODUCTION 8

Graphs: Definition -- Graphs as models -- Matrices and isomorphism; Paths, Cycles and Trails: Connections in graphs -- Bipartite graphs -- Eulerian circuits; Vertex Degrees and Counting: Counting and bijections -- Extremal problems; Directed Graphs: Definitions -- Vertex Degrees -- Eulerian graphs -- Orientation and tournaments.

UNIT II TREES ANDMATCHING 9

Basic Properties: Properties of trees -- Distance in trees and graphs; Spanning Trees and Enumeration: Enumeration of trees -- Spanning trees in graphs; Optimization and Trees: Minimum spanning trees -- Shortest paths -- Trees in Computer Science; Matchings and Covers: Maximum matching -- Hall's matching condition -- Min-Max theorems -- Independent sets and covers; Algorithms: Maximum bipartite matching -- Weighted bipartite matching -- Stable matchings.

UNIT III CONNECTIVITY AND COLORING 9

Cuts and Connectivity: Connectivity -- Edge connectivity -- Blocks; K-Connected Graphs: 2-Connected Graphs -- Connectivity of digraphs -- k-Connected and K-Edge-Connected Graphs; Network Flow Problems: Maximum network flow --Integral Flows; Vertex Colorings and Upper Bounds: Definitions --Upper bounds -- Brooks' theorem; Structure of k-Chromatic Graphs: Graphs with large chromatic number -- Extremal problems and Turan's theorem.

UNIT IV PLANARITY, EDGES AND CYCLES 8

Embeddings and Euler's Formula: Drawings in the plane -- Dual graphs -- Euler's formula; Characterization of Planar Graphs: Kuratawoski's theorem -- Convex embeddings -- Planarity testing; Line Graphs and Edge Colorings: Edge coloring -- Characterization of line graphs; Hamiltonian Cycles: Necessary and sufficient conditions -- Cycles in digraphs; Planarity, Coloring & Cycles: Tate's theorem -- Grinberg's theorem.

UNIT V ADVANCED TOPICS 10

Matroids: Properties of matroids -- Dual of a matroid -- Matroid operations; Ramsey Theory: Ramsey's theorem -- Ramsey numbers -- Graph Ramsey theory; Random Graphs: Existence and expectation -- Properties of almost all graphs -- Threshold functions -- Martingales; Eigenvalues of Graphs: The Characteristic polynomial -- Eigenvalues and graph Parameters -- Eigenvalues of regular graphs -- Eigenvalues and expanders.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Write precise mathematical definitions of objects in graph theory (K2)
- Read and analyse proofs on Trees and Bipartite Graph matching (K3)

- Read and analyze proofs on Graph Coloring and Connectivity (K3)
- Read and analyse proofs on Graph Planarity (K3)
- Read and analyse proofs on Matroids, Ramsey Theory, Graph Eigenvalues (K3).

TEXTBOOKS

1. Gary Chartrand, Ping Zhang, "Introduction to Graph Theory", McGraw Hill Education, 2011.

REFERENCE BOOKS

1. Douglous West, "Introduction to Graph Theory", Pearson Education, 2nd Edition,2015.
2. Narsingh Deo, "Graph Theory:With Application to Engineering and Computer Science", PrenticeHall of India,2003.
3. Frank Harary, "Graph Theory", Narosa Publishing House, 2001.
4. Adrian Bondy, U S R Murty, "Graph Theory", Springer, 2011.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1821	BIG DATA ANALYTICS	3	0	0	3

OBJECTIVES

- To know the fundamental concepts of big data and analytics
- To explore tools and practices for working with big data
- To learn about stream computing
- To know about the techniques that requires the integration of large amounts of data.

UNIT I INTRODUCTION TO BIG DATA 9

Evolution of big data -- Best practices for big data analytics -- Big data characteristics; Validating -- Promotion of the value of big data -- Big data use cases -- Characteristics of big data applications -- Perception and quantification of value; Big Data Tools and Techniques: Understanding big data storage -- General overview of high-performance architecture -- HDFS -- MapReduce and YARN; Map Reduce programming model; Review of basic data analytic methods using R.

UNIT II REGRESSION AND CLASSIFICATION 9

Advanced analytical theory and methods -- Regression: Linear regression -- Logistic regression; Classification: Decision trees -- Overview of a decision tree -- Decision treealgorithms -- Evaluating a decision tree -- Decision trees in R -- Naïve Bayes -- Bayes 'theorem -- Naïve Bayes classifier in R.

UNIT III DATA STREAM ANALYSIS 9

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 -- Case studies: Real time sentiment analysis, Stock market predictions.

UNIT IV FREQUENT ITEMSETS AND CLUSTERING 9

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 -- Clustering high dimensional data.

UNIT V NoSQL DATA MANAGEMENT FOR BIG DATA 9

NoSQL databases: Schema-less models -- Increasing flexibility for data manipulation -- Key value stores -- Document stores -- Tabular stores -- Object data stores -- Graph databases; Hive; Sharding; HBase; Case Study: Analyzing big data with twitter -- Big data for E-Commerce Big data for blogs.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Work with big data tools and its analysis techniques (K3)

- Analyze data by utilizing regression and classification algorithms (K4)
- Perform analytics on data streams (K3)
- Apply different mining algorithms and clustering techniques on big data (K3)
- Work with NoSQL databases and management (K2).

TEXTBOOKS

1. David Loshin, "Big Data Analytics: From Strategic Planning to Enterprise Integration withTools, Techniques, NoSQL, andGraph", Morgan Kaufmann / Elsevier Publishers, 2013.
2. Anand Rajaraman, Jeffrey DavidUllman,"Mining of MassiveDatasets", Cambridge University Press, 2012.

REFERENCE BOOKS

1. EMC Education Services, "Data Science and Big Data Analytics:Discovering, Analyzing, Visualizing and Presenting Data", Wiley Publishers, 2015. (Unit -II)
2. Bart Baesens, "Analytics in a Big Data World:The EssentialGuideto Data Science and its Applications", Wiley Publishers, 2015.
3. Kim H Pries,RobertDunnigan, "Big DataAnalytics:A PracticalGuide for Managers", CRC Press, 2015.
4. Jimmy Lin, Chris Dyer, "Data-Intensive Text Processing with MapReduce", Synthesis Lectures onHumanLanguage Technologies, Vol.3, No.1, Pages 1-177, Morgan Claypool publishers, 2010.
5. ArshdeepBahga, Vijay Madisetti, "Big Data Science & Analytics:A Hands- On Approach", VPT, 2016.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1822	AGILE METHODOLOGIES	3	0	0	3

OBJECTIVES

- To understand the basic concepts of Agile software process
- To learn about agile requirement engineering
- To learn planning and management in agile software development
- To study various agile methods
- To learn the principles of agile testing and quality assurance.

UNIT I INTRODUCTION 9

Iterative and Evolutionary Development; Introduction to Agile: Agile development -- Classification of methods -- Agile manifesto and principles -- Communication and feedback -- Specific agile methods -- Agile modelling; Theories for Agile Management; Management Accounting for Systems; Agile Project Management: Traditional versus RAD model for project management -- Task planning and effort tracking -- The project manager's new work.

UNIT II REQUIREMENTS ENGINEERING FOR AGILE METHODS 9

Traditional and Agile Requirement Engineering; Methods and Tools for Agile Practitioners:Requirements representation and documentation -- Requirements analysis -- Requirements management; Agile Approaches to Requirements Engineering: The customer -- Requirements evolution -- Non-functional requirements; Tools for Requirements Management in AMs.

UNIT III AGILEPROJECT PLANNING AND DEVELOPMENT MANAGEMENT 9

Agile Project Planning: The Project buffer and its usage -- Logical collection of inventories -- Critical path -- Parallel path -- Critical chain -- Project tracking metrics; Agile Development Management: Identifying and monitoring the flow -- Bottleneck; Agile Maturity Model: A new maturity model.

UNIT IV AGILE METHODS 9

Scrum: Method overview -- Life cycle -- Work products -- Values -- Roles and practices -- Process mixtures -- Adoption strategies; Extreme Programming; Unified Process; EVO.

UNITV AGILE TESTING AND QUALITY ASSURANCE 9

Agile testing: Nine principles and six concrete practices for testing on agile teams; Agile Metrics: Feature driven development (FDD) -- Financial and production metrics in FDD -- Agile approach to quality assurance -- Test driven development; SMM: A process improvement framework for agile requirements engineering practices.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Understand evolutionary, iterative and adaptive development methods (K2)

- Apply agile software process in requirement engineering (K3)
- Understand agile methods for project planning and development (K2)
- Apply agile methods for software design (K3)
- Apply agile based testing and quality assurance (K3)

TEXTBOOKS

1. DavidJ Anderson, Eli Schragenheim, "AgileManagement for Software Engineering: Applying the Theoryof Constraints for BusinessResults", PrenticeHall, 2003. (Unit 1, Unit 3, Unit 5)
2. CraigLarman, "AgileandIterative Development: A Manager 's Guide", Addison-Wesley, 2004. (Unit 1, Unit 4)
3. Elisabeth Hendrickson, "AgileTesting" Quality Tree SoftwareInc 2008. (Unit 5)

REFERENCE BOOKS

1. Hazza, Dubinsky, "Agile Software Engineering, Series: Undergraduate Topics in Computer Science", Springer, 2009. (Unit 5)
2. Chetankumar Patel, Muthu Ramachandran, "Story CardMaturity Model (SMM): A ProcessImprovement Framework for Agile Requirements Engineering Practices", Journal of Software, Academy Publishers, Vol 4, No5, 422-435, Jul 2009. (Unit 5)
3. Kevin C Desouza, "Agile Information Systems: Conceptualization, Construction, and Management", Butterworth-Heinemann, 2007.
4. Didar Zowghi, ZhiJin, "Requirements Engineering", Springer, chapter 15, 2014. (Unit 2)
5. Aybuke Aurum, Claes Wohlin, "Engineering and Managing Software Requirements", Springer 2005, chapter 14. (Unit 2)

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1823	NATURAL LANGUAGE PROCESSING	3	0	0	3

OBJECTIVES

- To learn language models
- To understand the levels of knowledge in language processing
- To develop NLP applications.

UNIT I OVERVIEW ANDLANGUAGE MODELING 8

Origins and challenges of NLP -- Knowledge in language processing -- NLP applications; Language Modeling: Language and grammar -- Grammar-based language models -- Lexical functional grammar -- Government and binding; Statistical Language Model: N-gram model -- Smoothing techniques.

UNIT II WORD LEVEL AND SYNTACTIC ANALYSIS 10

Word Level Analysis: Regular expressions -- Survey of morphology -- Word and sentence tokenization -- Stemmer -- Word classes -- Part-of-Speech Tagging: HMM POS tagging; Syntactic Analysis: Constituency -- Context-free grammar -- Dependency Grammar; Parsing: Top-down -- Bottom-up -- Ambiguity -- Early algorithm -- CYK -- Probabilistic CFG -- Probabilistic CYK parsing; Tree banks.

UNIT III SEMANTIC ANALYSIS 9

The representation of Meaning: Meaning representation -- Computational desiderata for representation; Lexical Semantics: Word senses -- Relations -- WordNet -- Thematic roles -- Selectional restrictions; Word Sense Disambiguation: Dictionary-based -- Supervised -- Minimally-supervised -- Unsupervised; Word Similarity: Thesaurus methods -- Distributional methods.

UNIT IV DISCOURSE PROCESSING, IR AND IE 9

Discourse Processing: Reference resolution -- Anaphora resolution algorithms -- Co-reference resolution; Information Retrieval: The vector space model -- Term weighting -- Evaluation of IR; Information Extraction: Named entity recognition -- Relation detection and classification.

UNIT V MACHINE TRANSLATION AND QUESTION ANSWERING 9

Machine Translation (MT): Problems in machine translation -- Classical MT -- Statistical MT; Factoid Question Answering: Question processing -- Passage retrieval -- Answer processing -- Evaluation of factoid answers.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Describe the language models (K3)
- Explain levels of knowledge in language processing (K3)
- Apply computational methods in semantic and discourse processing (K3)
- Apply NLP techniques to MT, IR, IE, QA and Summarization systems (K2)

- Apply evaluation metrics for different NLP applications (K3).

TEXTBOOKS

1. Daniel Jurafsky and James H Martin, "Speech and Language Processing: An introduction to Natural Language Processing, Computational Linguistics and Speech Recognition", 2nd Edition,PrenticeHall, 2008.
2. Tanveer Siddiqui, U S Tiwary, "Natural Language Processing and Information Retrieval", Oxford University Press, 2008.

REFERENCE BOOKS

1. Christopher D Manning, Hinrich Schutze, "Foundations of Statistical Natural Language Processing", MIT Press, 1999.
2. Nitin Indurkhya, Fred J Damerau, "Handbook of Natural Language Processing",2nd Edition,CRC Press, 2010.
3. StevenBird,EwanKlein,"Natural Language Processing withPython", O'Reilly Media,2009.
4. RuslanMitkov, "The Oxford Handbook of Computational Linguistics", OxfordUniversity Press, 2009.
5. NLTK – Natural Language Tool Kit - http://www.nltk.org/.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1824	ROBOTICS	3	0	0	3

OBJECTIVES

- To understand the basic concepts associated with the design, functioning, applications and social aspects of robots
- To study about the electrical drive systems and sensors used in robotics for various applications
- To learn about analyzing robot kinematics, dynamics through different methodologies and study various design aspects of robot arm manipulator and end-effector
- To learn about various motion planning techniques and the associated control architecture
- To understand the implications of AI and other trending concepts of Robotics.

UNIT I FOUNDATION 9

Introduction -- Brief history -- Definition -- Anatomy -- Types -- Classification -- Specification and need based applications -- Role and need of robots for the immediate problems of the society -- Future of mankind and automation-ethical issues -- Industrial scenario local and global -- Case studies on mobile robot research platform and industrial
serial arm manipulator.

UNIT II BUILDING BLOCKS OF A ROBOT 9

Types of electric motors : DC -- Servo -- Stepper; Specification -- Drives for motors -- Speed & direction control and circuitry -- Selection criterion for actuators -- Direct drives -- Non-traditional actuators - Sensors for localization -- Navigation -- Obstacle avoidance and path planning in known and unknown environments -- Optical -- Inertial -- Thermal -- Chemical -- Biosensor -- Other common sensors -- Case study on choice of sensors and actuators for maze solving robot and self-driving cars.

UNIT III EMATICS, DYNAMICSAND DESIGN OF ROBOTS & END-EFFECTORS 9

Robot kinematics -- Geometric approach for 2R, 3R manipulators -- homogenous transformation using D-H representation -- Kinematics of WMR -- Lagrangian formulation for 2R robot dynamics -- Mechanical design aspects of a 2R manipulator, WMR -- End-effector: Common types and design case study.

UNIT IV NAVIGATION, PATH PLANNING AND CONTROL ARCHITECTURE 9

Mapping & Navigation -- SLAM, Path planning for serial manipulators -- Types of control architectures -- Cartesian control -- Force control and hybrid position/force control -- Behavior based control -- Application of Neural network, fuzzy logic, optimization algorithms for navigation problems -- Programming methodologies of a robot.

UNIT V AI AND OTHER RESEARCH TRENDS IN ROBOTICS 9

Application of Machine learning -- AI -- Expert systems -- Tele-robotics and Virtual reality -- Micro & Nanorobots -- Unmanned vehicles -- Cognitive robotics -- Evolutionary robotics -- Humanoids.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Understand the concepts of industrial robots (K2)
- Examine different sensors and actuators for applications like maze solving and self-driving cars (K2)
- Design a 2R robot & an end-effector and solve the kinematics and dynamics of motion for robots (K3)
- Understand the navigation and path planning techniques for robot motion planning (K2)
- Understand the impact and progress of AI in the field of robotics (K2).

TEXTBOOKS

1. Saeed B Niku, “Introduction to Robotics, Analysis,System, Applications”, Pearson educations, 2002.
2. Roland Siegwart, Illah Reza Nourbakhsh, “Introduction to Autonomous Mobile Robots”, MIT Press, 2011.

REFERENCE BOOKS

1. Richard David Klafter, Thomas A Chmielewski, Michael Negin, “Robotic engineering: An Integrated Approach”, PrenticeHall, 1989.
2. Craig,J J, “Introduction to Robotics: Mechanics and Control”, 2ndEdition, Addison-Wesley, 1989.
3. K S Fu, R C Gonzalez andC S G Lee, “Robotics:Control, Sensing,Vision and Intelligence”, McGraw-Hill, 1987.
4. WesleyE SnyderR, “Industrial Robots,Computer Interfacing andControl”, PrenticeHall International Edition,1988.
5. Robin Murphy, “Introduction to AI Robotics”,MIT Press, 2000.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1825	NETWORK AND SERVER SECURITY	3	0	0	3

OBJECTIVES

- To study about the essentials of computer security
- To acquire knowledge on TCP/IP security, firewalls, IPSec, Virtual Private Networks, and intrusion detection systems
- To understand how various security mechanisms work, and correlate These security mechanisms with security principles
- To learn the security aspects of data center
- To learn the security protocols and technologies with respect to infrastructure.

UNIT I INTRODUCTION 9

Computer Security Concepts -- Security Attacks -- Security Services - Security Mechanisms - A Model for Network Security- Standards. Attack on Public Key Cryptography -- Public Key Certificates: X.509 Authentication services -- Attacks on PKI -- Types of Digital Certificates.

UNIT II SECURITY PRACTICES & SYSTEM SECURITY 9

Internet Firewalls for Trusted System: Roles of Firewalls -- Types of Firewalls -- Netfilter -- IPtables -- Firewall design Principles -- DNS Attacks -- Cache Poisoning -- SET (Secure Electronic Transaction) for E-Commerce Transactions. Intruder -- Intrusion detection system -- Virus and related threats -- Countermeasures.

UNIT III E-MAIL, IP & WEB SECURITY 9

E-mail Security: Security Services for E-mail-attacks possible through E-mail -- Establishing keys privacy authentication of the source -- Message Integrity-Non-repudiation -- Pretty Good Privacy -- S/MIME; IPSecurity: Overview of IPSec -- IP and IPv6 -- Authentication Header -- Encapsulation Security Payload (ESP) -- Internet Key Exchange; Web Security: SSL/TLS Basic Protocol -- Computing the keys -- Client authentication -- PKI as deployed by SSLAttacks fixed in v3 -- Exportability-Encoding -SET.

UNIT IV DATA CENTER SECURITY OVERVIEW 9

Data center security overview: Need for a secure data center -- Vulnerabilities and common attacks; Network Security Infrastructure; Security Fundamentals; Data center security frameworks: Security policies -- Security lifecycle; Secure Management Framework.

UNIT V SECURITY PROTOCOLS ANDTECHNOLOGIES 9

Security Protocols and Technologies: Cryptography -- PKI -- Transport Security -- Authentication Protocols and Technologies; Network management security; Integrating security into the infrastructure: Defining security zone -- Internet Edge -- Intranet Server Farm -- Server-Farm Design Alternative -- Management Network.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

• Understand the essentials of computer security (K2)

• Understand various applications of network security (K2)
• Understand the essentials of security practices and system security (K2)
• Explain the security aspects in designing a datacenter (K2)
• Designthe security protocols for the infrastructure (K3)

TEXTBOOKS

1. WilliamStallings,"Network SecurityEssentials:Applications andStandards", Sixth Edition,Pearson, 2017.
2. Mauricio Arregoces, Maurizio Portolani, "Data Center Fundamentals", Cisco Press, 2003.

REFERENCE BOOKS

1. Charlie Kaufman, Radia Perlman and Mike Speciner, "Network Security: Private Communication in a Public World", Second Edition,Pearson Education, 2017.
2. Wenliang Du, "Computer Security:A Hands-on Approach", CreateSpace Independent Publishing Platform, First Edition,2017.
3. Colin Boyd and Anish Mathuria, "Protocols for Authentication and Key Establishment (Information Security and Cryptography), Springer, 2003.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1826	FORMAL SYSTEM VERIFICATION	3	0	0	3

OBJECTIVES

- To understand the need of logics for specification and verification of computer systems
- To learn program correctness using Hoare Logic
- To learn the skill of writing formal specifications in LTL and CTL
- To learn model checking algorithms for LTL and CTL
- To learn model checking for Timed Systems.

UNIT I FOUNDATIONS 9

Propositional Logic: Syntax -- Semantics -- Natural deduction; Predicate Logic: Syntax -- Semantics -- Natural deduction; Linear-time Temporal Logic: Syntax -- Semantics -- Specifications in LTL; Branching-time Logic: Syntax -- Semantics -- Specifications in CTL; CTL* and the expressive powers of LTL and CTL.

UNIT II HOARE LOGIC AND PROGRAM CORRECTNESS 9

A Framework for Program Correctness: A core programming language -- Hoare triples -- Partial and total correctness -- Program variables and logical variables; Proof Calculus for Partial Correctness: Proof rules -- Proof tableaux; Proof Calculus for Total Correctness.

UNIT III MODEL CHECKING 9

Model-checking Algorithms: The CTL model-checking algorithm -- CTL model checking with fairness -- The LTL model-checking algorithm; CTL* Model-checking Algorithm; Model Checking using Automata; Checking Emptiness; Translating LTL into Automata; On-the-fly Model checking.

UNIT IV SYMBOLIC MODEL CHECKING 9

Binary Decision Diagrams: Representing Boolean formulas -- Representing Kripke structures; Fixpoint Representations; Symbolic Model Checking for CTL; Fairness in Symbolic Model Checking; Counterexamples and Witnesses; Relational Product Computations; Symbolic Model Checking for LTL; NuSMV.

UNIT V MODEL CHECKING TIMED SYSTEMS 9

Timed Automata: Semantics -- Time divergence -- Timelock -- Zenoness; Timed Computation Tree Logic; TCTL Model Checking: Eliminating timing parameters -- Region transition systems -- The TCTL model-checking algorithm; Model checkers for Timed Automata: UPPAAL – KRONOS.

TOTAL PERIODS: 45

OUTCOMES

On successful completion of this course, the student will be able to

- Explain, analyze and compare various model checking algorithms (K3)
- Prove partial correctness of simple programs using Hoare logic (K3)
- Write formal properties and specifications in CTL and LTL (K3)
- Specify and verify simple systems using NuSMV (K3)

- Specify and verify simple systems using UPPAAL (K3).

TEXTBOOKS

1. M Huth, M Ryan, "Logic in Computer Science – Modeling and Reasoning About Systems", 2nd Edition, Cambridge University Press,2004. (Units I, II and III)
2. Edmund Clarke, Orna Grumberg, Doron Peled, "Model Checking", The MIT Press, 1999. (Units III and IV)
3. C Baier, J Katoen, "Principles of Model Checking", The MIT Press,2008. (Unit V)

REFERENCE BOOKS

1. Michael Clarke, Thomas Henzinger, Helmut Veith, Roderick Bloem, "Hand- book of Model Checking", Springer 2018
2. Orna Grumberg, Helmut Veith,"25 Years of ModelChecking:History, Achievements, Perspectives" Springer-Verlag, 2008
3. Zohar Manna, Amir Pnueli, "Temporal Verification of Reactive Systems: Safety", Springer-Verlag, 2012
4. Krzysztof R. Apt,FrankS. de Boer, Ernst-Rüdiger Olderog, "Verification of Sequential and Concurrent Programs", Springer, 3rd edition, 2009.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1941	MACHINE LEARNING APPLICATIONS	2	0	2	3

OBJECTIVES

- To understand the need and types of machine learning techniques for various problems
- To study the various supervised learning algorithms in machine learning
- To choose appropriate machine learning algorithms to solve realistic problems.

UNIT I MACHINE LEARNING LANDSCAPE 5

Machine learning and it's use; Types of machine learning; Challenges of ma- chine learning; Testing and validating.

UNIT II DEVELOPING A MACHINE LEARNING APPLICATION 6

Working with real data– Look at the big picture – Get the data– Discover and visualize the data– Data preparation – Select and train model.

UNIT III CLASSIFICATION ANDREGRESSION 8

Classification: Training a binary classifier – Performance measures – Multiclass classification; Regression: Linear regression – Gradient descent – Logistic Regression; Support Vector Machines :Linear SVM classification – Nonlinear SVM classification.

UNIT IV TREE MODELS ANDENSEMBLE LEARNING 6

Decision Trees: Training and visualizing trees – Making predictions – Estimating class probabilities – CART training algorithm – Regularization of hyperparameters; Ensemble learning: Voting classifiers– Bagging– Random forests– Boosting.

UNIT V ARTIFICIAL NEURAL NETWORKS 5

From Biological to Artificial Neurons: Biological neurons – Logical computations with neurons – Perceptron – Multi-Layer Perceptron and backpropagation; Training a MLP network – Fine tuning neural network hyper parameters; Introduction to Deep Learning.

TOTAL PERIODS (THEORY): 30

SUGGESTIVE EXPERIMENTS

1. Data analysis
2. Machine learning application for house price prediction
3. Classification of Iris dataset using multiclass classification
4. Loan amount prediction using linear regression
5. E-mail spam detection using support vector machine
6. Predicting Diabetes using decision tree
7. Handwritten character recognition using neural networks.

TOTAL PERIODS (PRACTICAL): 30
TOTAL PERIODS: 60

OUTCOMES

On successful completion of this course, the student will be able to

- Understand the basic concepts and types of machine learning (K2)
- Understand the various steps in developing a machine learning application (K2)
- Apply various algorithms for classification and regression tasks (K3)
- Apply tree and ensemble models for various problems (K3)
- Apply the neural network algorithm for real world problems (K3).

TEXTBOOKS

1. Aurelien Geron, "Hands-On Machine Learning with Scikit-Learn and Tensor Flow", O'Reilly Media,2017.

REFERENCE BOOKS

1. Stephen Marsland, "Machine Learning – An Algorithmic Perspective",2nd Edition, Chapman and Hall/CRC Machine Learning and Pattern Recognition Series, 2014.
2. Jason Bell, "Machine learning – Hands on for Developers and Technical Professionals", 1st Edition, Wiley, 2014.
3. Richert, Willi, "Building Machine Learning Systems with Python", Packt Publishing Ltd, 2013.
4. TomM. Mitchell, "Machine Learning", McGraw-Hill Education (India) Private Limited, 2013.
5. Andreas C. Muller, Sarah Guido, "Introduction to Machine Learning with Python", O'Reilly Media,2016.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1942	WEB TECHNOLOGY	2	0	2	3

OBJECTIVES

- Learn to design static web pagesusingHTML5 and CSS3
- Learn to develop client-side scripts using JavaScript
- Learn to write server-side scriptsusingNode.js
- Learn to develop web application using ExpressJS

UNIT I WEB PAGE DESIGNING USING HTML 6

Introduction: HTML Basics – Document Structure – Formatting Texts – Lists and Backgrounds – Hyperlinks and Anchors; Page Layout and Navigation: Layouts – Tables – Forms – Sound and Videos.

UNIT II STYLING WEBPAGE WITH CSS3 6

CSS Basics: HTML for CSS – Creating Styles and Style Sheets – Selectors; Ap- plied CSS: Formatting Texts – Margin, Padding and Borders – Adding Graphics to web pages– Navigation bars and links – Formatting Tables and Forms.

UNIT IIII CLIENT-SIDE JAVA SCRIPT 6

Introduction to Java script: JavaScript usage– Role in Web page – Role in Web Development; Script setup; Thebuilding Blocks: DataTypes, Literals, Variables; Dialogbox; Operators; Conditionals & loops; Functions; Objects; Core Objects; working with images; Handling Events.

UNIT IV SERVER SIDE JAVASCRIPT WITH NODE AND EXPRESS 6

Introduction to Express; Getting started with Node; Saving time with Express: Scaffolding – views and layouts – Static files and views– Dynamic content in views; Request and Response Objects; Templating with Handlebars.

UNIT V BUILDINGWEB APPLICATIONS 6

Form Handling: Sending Client data to Server– HTML forms– Different approaches for form handling – Form Handling with Express; Persistence; Routing; Static Content; Debugging.

TOTAL PERIODS(THEORY): 30

SUGGESTIVE EXPERIMENTS

1. Design a Personal website using HTML5 elements.
2. Design a web page using form controls and perform in-built validation (e.g. Login form, Registration form)
3. Use Embedded, Inline and External Style sheets for styling a web page
4. Use Selectors to format texts, margins, padding, borders, tables, forms for a web page
5. Develop Java script programs using event handling mechanism (e.g. Calculator, Tic Tac Toe game)
6. Develop a web application for personal website using ExpressJS
7. Develop a web application for student information system that allows to perform CRUD operations using ExpressJS and MySQL

TOTAL PERIODS(PRACTICAL): 30
TOTAL PERIODS: 60

OUTCOMES

On successful completion of this course, the student will be able to

- Design static web pagesusingHTML5 (K3)
- Style the web pagesusingCSS3 (K3)
- Develop client-side script susing JavaScript(K3)
- Create server-side scriptsusingNode.js(K3)
- Develop web applications using ExpressJS (K3).

TEXTBOOKS

1. FaitheW empen, "StartHere!Learn HTML5",Microsoft,O'Reilly, 2013. (Unit I)
2. David McFarland, "CSS3: The Missing Manual", 3rd Edition, O'Reilly, December 2012. (Unit II)
3. Ellie Quigley, "JavaScript by example", 2nd Edition, Prentice Hall, 2011. (Unit III)
4. Ethan Brown, "Development with Node and Express", O'Reilly, 1st Edition, July 2014. (Unit IV, Unit V)

REFERENCE BOOKS

1. Jennifer Niederst Robbins, "Learning Web Design", O'Reilly, 4th Edition, 2012.
2. Robert W Sebesta, "Programming theWorldWideWeb",4th Edition, Pearson Education, 2007.
3. Shelly Powers, "Learning JavaScript", O'Reilly, 2nd Edition,2008.
4. Basaret Ali Syed, "Beginning Node.js", Apress, December 2014.
5. AzatMardan, "Express js Guide", Lean Publishing, 2014.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1041	INTRODUCTION TO DATASTRUCTURES	2	0	2	3

OBJECTIVES

- To understand the concepts of sorting and searching techniques
- To comprehend the notion of linear and nonlinear data structures
- To use a suitable data structure for the given problem and implement them in a high-level programming language.

UNIT I SORTING AND SEARCHING TECHNIQUES 6

Sorting: Insertion sort -- Merge sort -- Quicksort; Searching: Linear search -- Binary search.

UNIT II LIST 6

List -- Array implementation -- Linked lists -- Applications of Lists: Polynomial addition.

UNIT III STACK AND QUEUE 6

Stack: Array implementation of stack -- Applications: Evaluating postfix expressions; Queue: Array implementation of queues -- Applications of queues.

UNIT IV TREES 6

Tree: Preliminaries -- Tree traversals -- Binary Search Trees -- Binary Heaps -- Applications of trees.

UNIT V GRAPHS 6

Graph algorithms: Definitions -- Representation of graphs -- Breadth first traversal -- Depth first traversal -- Shortest-path algorithm: Dijkstra's algorithm.

TOTAL PERIODS(THEORY): 30

SUGGESTIVE EXPERIMENTS

1. Implement insertion sort and merge sort
2. Implement linear search and binary search.
3. Implement List ADT using arrays.
4. Represent polynomial as a linked list and perform polynomial addition
5. Array implementation of stack ADT
6. Evaluate postfix expressions using Stack ADT
7. Produce the pre-order, in-order, and post-order traversals of binary trees.
8. Represent graph using adjacency matrix and perform depth first traversal on the graph
9. Implement Dijkstra’s algorithm to find the shortest path using graph

TOTAL PERIODS(PRACTICAL): 30

TOTAL PERIODS: 60

OUTCOMES

On successful completion of this course, the student will be able to

- Implement various sorting and searching algorithms (K2)
- Implement list and solve problems using them (K3)

- Implement stack and queue and use them to solve problems (K3)
- Implement trees and use them to solve problems (K3)
- Implement graphs and use them to solve problems (K3).

TEXTBOOKS

1. M A Weiss, “Data Structuresand Algorithm Analysis in C”, Second Edition, Pearson Education, 2006.
2. Richard F Gilberg, Behrouz A Frouzan, “Data Structures: A Pseudocode Approach with C”, 2nd Edition,Cengage India,2007.

REFERENCE BOOKS

1. V Aho, J E Hopcroft, and J D Ullman, “Data Structures and Algorithms”, Pearson Education, 1st Edition, 2003.
2. Ellis Horowitz, Sartaj Sahni, Susan Anderson-Freed, “Fundamentals of Data Structures in C”, 2nd Edition,University Press, 2008
3. S Sridhar, “Design and Analysis of Algorithms”, 1st Edition, Oxford University Press. 2014.
4. Byron Gottfried, Jitender Chhabra, “Programming with C” (Schaum’s Out- lines Series), McGraw-Hill Higher Ed,3rd Edition,2010.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1042	OBJECT ORIENTED PROGRAMMING TECHNIQUES	2	0	2	3

OBJECTIVES

- To learn the basics of Object-Oriented Programming
- To learn the programming constructs of Java
- To know the principles of inheritance and polymorphism.

UNIT I INTRODUCTION **6**

Principles of OOP: Classes -- Objects -- Data hiding -- Data encapsulation -- Inheritance -- Polymorphism; Definition of classes: objects - methods -- Access specifiers -- Static and final classes and members; Object construction and destruction.

UNIT II BASICS OF JAVA **6**

Features of Java -- Basics of Java programming: Data types - Variables -- Operators -- Control structures -- Arrays -- Strings.

UNIT III CLASSES AND OBJECTS **6**

Objects and Classes in Java: Defining classes -- Methods -- Access specifiers -- Static members; Constructors: Constructor overloading; Method overloading -- Recursion -- Passing and returning object from method -- Packages in Java.

UNIT IV INHERITANCE **6**

Inheritance: Definition -- Types of inheritance: Single -- multilevel - multiple -- hierarchical -- hybrid; Subclass constructors -- Interfaces in Java: Definition, Implementation -- Extending interfaces -- Examples in Java.

UNIT V POLYMORPHISM **6**

Polymorphism: Method overloading and overriding -- Dynamic method dispatch; Exceptions: Hierarchy, Built-in exceptions, Creating own exception; Definition and Concepts: Generic methods -- Generic types.

TOTAL PERIODS(THEORY): 30

SUGGESTIVE EXPERIMENTS

1. Program in Java using classes and objects (Eg: Marklist generation)
2. Program to perform string operations using ArrayList.
3. Program to implement packages (Eg: Currency converter)
4. Program using Inheritance (Eg: Payroll application)
5. Program to implement Polymorphism (Eg. Area of different shapes)
6. Program to implement generic methods (Eg: Sorting)
7. Design a Java interface for ADTs (Eg. Stack, Queue).

TOTAL PERIODS(PRACTICAL): 30

TOTAL PERIODS: 60

OUTCOMES

On successful completion of this course, the student will be able to

- Understand object-oriented programming features (K2)
- Develop programs using basic constructs of Java (K3)
- Solve problems using classes and methods (K3)
- Develop programs using inheritance and interfaces (K3)
- Use the concept of polymorphism (K3).

TEXTBOOKS

1. Herbert Schildt,"Java:The Complete Reference", 8th Edition,McGraw Hill Education, 2011.

REFERENCE BOOKS

1. Timothy Budd, "Understanding Object-oriented Programming with Java", Updated Edition,Pearson Education, 2000.
2. C ThomasWu,"An introduction to Object-oriented Programming with Java", Fourth Edition,Tata McGraw-Hill Publishing company Ltd., 2006.
3. Cay S Horstmann, Garycornell, "Core Java Volume– I Fundamentals" 9th Edition, Prentice Hall, 2013.
4. PaulDeitel, Harvey Deitel, "JavaSE 8 for Programmers", 3rdEdition, Pearson, 2015.
5. Steven Holzner, "Java 2 Black Book", Dreamtech Press, 2011.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1043	PROBLEM SOLVING AND PROGRAMMING IN C	2	0	2	3

OBJECTIVES

- To use basic programming concepts of C
- To develop programs using loops, functions, arrays, pointers, strings, structures and files.

UNIT I BASICS **6**

Algorithm – Structure of a C program – DataTypes:built-in,user-defined – Variables and constants – Input and output statements – Operators, expressions, and assignment statements – Control statements.

UNIT II FUNCTIONS **6**

Function prototype – Function definition and call – Passing parameters – Built- in functions – Recursion.

UNIT III ARRAYS AND POINTERS **6**

Arrays:Declaration, initialization – One-dimensional arrays – Multi-dimensional arrays;Pointers: Pointer operators – Pointer operations – Array of pointers – Passing arrays to functions.

UNIT IV STRINGS, STRUCTURES ANDUNIONS **6**

Strings:Constants and variables, reading and writing strings, string operations, string library;Command line arguments; Structures: Structures and functions– Array of structures – Pointer to a structure – unions;Dynamic memory allocation.

UNIT V FILES **6**

Files: opening and closing a datafile – Creating a datafile – Processing a data file – Unformatted datafiles – Formatted input and output – Line input and output – File error handling.

TOTAL PERIODS(THEORY): 30

SUGGESTIVE EXPERIMENTS

1. Program using, I/O statements, conditional constructs (if, switch, ternary) and looping constructs (for, while, do-while)
2. Program functions with different parameter passing techniques: Call by value, call by reference (e.g. changing the elements of an array)
3. Program one dimensional arrays (e.g. Insertion sort)
4. Program using strings and their operations (e.g. concatenation of strings, extracting a substring, checking for palindrome)
5. Program to demonstrate simple structure manipulations (e.g. generating a transcript with CGPA and class obtained)
6. Programs to demonstrate file operations (e.g.countthe number of characters, words and lines in a file).

TOTAL PERIODS(PRACTICAL): 30

TOTAL PERIODS: 60

OUTCOMES

On successful completion of this course, the student will be able to

- Understand the basic programming constructs in C (K2)
- Write functions in C (K2)
- Develop C programs using arrays and pointers (K3)
- Write programs using strings and structures (K3)
- Do operations with files in C (K2).

TEXTBOOKS

1. Byron Gottfried, "Programming with C", (Schaum's Outlines Series), McGraw- Hill Education, 3rd Edition,2017.
2. Brian W Kernighan, DennisM Ritchie, "The C Programming Language", Pearson Education India,2nd Edition,2015.

REFERENCE BOOKS

1. Reema Thareja, "Programming in C", Oxford University Press, 2nd Edition, 2016.
2. Yashwant Kanetkar, "Let Us C", BPB Publications, 14th Edition,2016.
3. King K N, "C Programming: A Modern Approach", W. W. Norton & Com- pany,2nd Edition,2008.
4. Herbert Schildt, "C The Complete Reference", McGraw-Hill Education, 4thEdition,2017.
5. Stephen G Kochan, "Programming in C", 3rd edition, Pearson Ed, 2004.

COURSE CODE	COURSE TITLE	L	T	P	C
UCS1044	INTRODUCTION TO BIG DATA ANALYTICS	2	0	2	3

OBJECTIVES

- To understand the competitive advantages of big data analytics
- To understand the distributed storage for big data
- To learn distributed method for processing of big data
- To understand how to represent unstructured data using NoSQL and processing
- To learn how statistical methods are used for analyzing big data

UNIT I INTRODUCTION TO BIG DATA 9

Introduction – Understanding Big Data– Big Data:Benefitting– Managing – Organizing and Analyzing Big Data: Learning and Analytics;Technology Challenges for Big Data.

UNIT II HDFS 9

Introduction – Distributed File System– Google File System– HDFS DesignGoals – Using HDFS.

UNIT III DATA PROCESSING USINGMAPREDUCE 10

Introduction – MapReduce Overview – Working of MapReduce – Programming– Writing and Testing MapReduce Programs.

UNIT IV NoSQL 9

Introduction to NoSQL – Characteristics of NoSQL – NoSQL Storage Types– Advantages and Drawbacks - NoSQL Database Framework: Hive and HBase.

UNIT V DATA ANALYSIS 10

Statistical Methods: Regression modelling – Multivariate analysis;Classification: SVM – Decision Trees; Linear Classifiers

TOTAL PERIODS(THEORY): 30

SUGGESTIVE EXPERIMENTS

Hadoop

1. Applications using Map-Reduce programming (Examples: word count/ frequency programs / matrixmultiplication)

R

1. Linear and logistic Regression (Loan prediction using Credit approval dataset, Sales prediction using Bigmart dataset)
2. SVM / Decision tree classification techniques (Flower type classification basedon available attributes usingIris dataset, Passengers survival classification usingtitanic dataset)
3. Clustering (Document categorization by multiclass techniques)
4. Visualizedatausingany plotting framework

Database

1. Application that stores data in HBase (Sentiment analysis using twitter dataset)

TOTAL PERIODS(PRACTICAL): 30
TOTAL PERIODS: 60

OUTCOMES

On successful completion of this course, the student will be able to

- Understand how to leverage the insights from big data analytics (K2)
- Understand and apply distributed computing for better storage of data (K3)
- Develop applications using Hadoop related tools(K4)
- Use database frameworks like Hive and HBase fordata analysis(K3)
- Solve applications using statistical and data analysis methods (K3).

TEXTBOOKS

1. Para Kulkarni, Sarang Joshi, “Big Data Analytics”, PHI Learning, 2016.
2. Anil Maheshwari, “Big Data Essentials”, McGraw-Hill, 2019

REFERENCE BOOKS

1. ArshdeepBahga, Vijay Madisetti, “Big Data Analytics:A Hands-On Approach”, Published by A Hands-on Approach Textbooks, 2016.
2. Bill Franks,“Taming the Big Data Tidal Wave:Finding Opportunities in Huge DataStreamswith Advanced Analytics”, John Wiley & sons, 2012.
3. Gaurav Vaish, “Getting Started with NoSQL”, Packt Publishing Ltd, 2013.
4. E Capriolo, D Wampler, J Rutherglen, “Programming Hive”,O’Reilly, 2012.
5. Lars George, “HBase: The Definitive Guide”, O’Reilly, 2011.