Sri Sivasubramaniya Nadar College of Engineering

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

Rajiv Gandhi Salai (OMR), Kalavakkam – 603110

Regulations 2018

Curriculum and Syllabi

for

Master of Engineering

in

Computer Science and Engineering

SRI SIVASUBRAMANIYA NADAR COLLEGE OF ENGINEERING, KALAVAKKAM – 603 110
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

VISION

To emerge as a world class technology department through education, innovation, and collaborative research.

MISSION

- To impart quality education to students
- To create and disseminate knowledge for the betterment of mankind.
- To establish a centre of excellence in collaboration with industries, research laboratories and other agencies to meet the changing needs of society.
- To provide individual attention and enable character building.
- To encourage entrepreneurship skills among students

M. E. Degree Programme in Computer Science and Engineering

Program Educational Objectives (PEOs)

PEO1: Graduates will pursue research or be successfully employed in academia / industries associated with Computer Science and Engineering, or become entrepreneurs, adapting to new technologies and engaging in continuous education and training.

PEO2: Graduates will take leadership roles, making holistic decisions guided by professional, ethical, societal, economic, legal, and environmental considerations, and communicating clearly with stakeholders.

Program Outcomes (POs)

The graduates of the M.E degree program in Computer Science and Engineering will be able to

PO1: Independently carry out research / investigation, and development work to solve practical problems (K6)

PO2: Read, write, and present a substantial technical report/document

PO3: Demonstrate a degree of mastery over the area as per the specialization of the program. The mastery should be at a level higher than the requirements in the appropriate bachelor program (K6)

PO4: Analyze, learn, and build models from real world data using machine learning techniques and deploy them (K5)

Mapping of POs to PEOs

POs	**PEOs**	
	PEO1: Graduates will pursue research or be successfully employed in academia / industries associated with Computer Science and Engineering, or become entrepreneurs, adapting to new technologies and engaging in continuous education and training.	PEO2: Graduates will take leadership roles, making holistic decisions guided by professional, ethical, societal, economic, legal, and environmental considerations, and communicating clearly with stakeholders.
PO1: Independently carry out research / investigation, and development work to solve practical problems	3	2
PO2: Read, write and present a substantial technical report/document	2	3
PO3: Demonstrate a degree of mastery over the area as per the specialization of the program. The mastery should be at a level higher than the requirements in the appropriate bachelor program	3	2
PO4: Analyze, learn, and build models from real world data using machine learning techniques and deploy them	3	2

Contribution 1: Reasonable 2: Significant 3: Strong

MAPPING OF COURSES TO PROGRAM OUTCOMES

	PO1	PO2	PO3	PO4
Semester I				
Applied Probability and Statistics	3		1	1
Advanced Data Structures and Algorithms	2		2	
Multicore Architecture and GPU Computing	2		2	
Advanced Software Engineering	2		2	
Machine Learning	2		2	2
Network Design and Technologies	2		2	
Advanced Data Structures and Algorithms Lab	2		2	
Ethical Practices		3	2	
Semester II				
Modern Operating Systems	2		2	
Internet of Things	2		2	2
Web Application Development	2		2	
Data Analytics	2		2	2
Professional Elective 1				
Professional Elective 2				
Data Analytics Lab	2	1	2	2
Mini Project	2	1	2	
Technical Paper Writing and Seminar	2	3	2	
Semester III				
Professional Elective 3				
Professional Elective 4				
Open Elective 1				
Project Work Phase I	3	3	3	3
Semester IV				
Project Work Phase II	3	3	3	3
Professional Elective I				
Cloud Computing	2		2	
Advanced Databases	2		2	
Web Engineering	2		2	
Natural Language Processing	2	1	2	2
Wireless Networks	2		2	
Principles of Programming Languages	2		2	
Introduction to Cryptography	2		2	
Professional Elective 2				
Parallel Programming	2		2	
Software Architecture and Design	2		2	
Image Processing and Analysis	2		2	2
Speech Processing and Synthesis	2		2	
Bio Inspired Computing	2		2	
Bio Informatics	2		2	
Advanced Algorithms	2		2	
Network Security	2		2	

Professional Elective 3				
Real Time Systems	2		2	
Software Project Management	2		2	
Computer Vision	2		2	
Information Retrieval Techniques	2	1	2	2
Data Visualization Techniques	2		2	
Software Defined Networks	2		2	
Logic in Computer Science	2		2	
Professional Elective 4				
Mobile and Pervasive Computing	2		2	
Agile Software Engineering	2		2	
Social Network Analysis	2		2	
Deep Learning	2		2	2
Formal Verification	2		2	
Storage and Server Security	2		2	

SRI SIVASUBRAMANIYA NADAR COLLEGE OF ENGINEERING, KALAVAKKAM

(An Autonomous Institution Affiliated to Anna University Chennai)

REGULATIONS 2018

CHOICE BASED CREDIT SYSTEM

M.E. COMPUTER SCIENCE AND ENGINEERING

CURRICULUM

Semester I

Sl. No	Course Code	Course Title	Category	Contact Periods	L	T	P	C
Theory								
1	PMA1177	Applied Probability and Statistics	FC	4	4	0	0	4
2	PCP1176	Advanced Data Structures and Algorithms	PC	3	3	0	0	3
3	PCP1101	Multicore Architecture and GPU Computing	PC	3	3	0	0	3
4	PCP1177	Advanced Software Engineering	PC	3	3	0	0	3
5	PCP1178	Machine Learning	PC	5	3	0	2	4
6	PCP1102	Network Design and Technologies	PC	4	2	0	2	3
Practical								
7	PCP1197	Advanced Data Structures and Algorithms Lab	PC	4	0	0	4	2
8	PCP1111	Ethical Practices	EEC	2	0	0	2	1
			Total	28	18	0	10	23

Semester II

Sl. No	Course Code	Course Title	Category	Contact Periods	L	T	P	C
Theory								
1	PCP1201	Modern Operating Systems	PC	5	3	0	2	4
2	PCP1276	Internet of Things	PC	4	2	0	2	3
3	PCP1202	Web Application Development	PC	3	3	0	0	3
4	PCP1281	Data Analytics	PC	3	3	0	0	3
5		Professional Elective 1	PE	3	3	0	0	3
6		Professional Elective 2	PE	3	3	0	0	3
Practical								
7	PCP1211	Data Analytics Lab	PC	4	0	0	4	2
8	PCP1217	Mini Project	EEC	2	0	0	2	1
9	PCP1215	Technical Paper Writing and Seminar	EEC	2	0	0	2	1
			Total	29	17	0	12	23

Semester III

Sl.No	Course Code	Course Title	Category	Contact Periods	L	T	P	C
Theory								
1		Professional Elective 3	PE	3	3	0	0	3
2		Professional Elective 4	PE	3	3	0	0	3
3		Open Elective 1	OE	3	3	0	0	3
Practical								
4	PCP1318	Project Work Phase I	EEC	12	0	0	12	6
			Total	21	9	0	12	15

Semester IV

Sl.No	Course Code	Course Title	Category	Contact Periods	L	T	P	C
Practical								
1	PCP1418	Project Work Phase II	EEC	24	0	0	24	12
			Total	24	0	0	24	12
							Grand Total	73

Professional Elective 1

Sl.No	Course Code	Course Title	Category	Contact Periods	L	T	P	C
1	PCP1277	Cloud Computing	PE	3	3	0	0	3
2	PCP1221	Advanced Databases	PE	3	3	0	0	3
3	PCP1222	Web Engineering	PE	3	3	0	0	3
4	PCP1278	Natural Language Processing	PE	3	3	0	0	3
5	PCP1222	Wireless Networks	PE	3	3	0	0	3
6	PCP1224	Principles of Programming Languages	PE	3	3	0	0	3
7	PCP1225	Introduction to Cryptography	PE	3	3	0	0	3

Professional Elective 2

Sl.No	Course Code	Course Title	Category	Contact Periods	L	T	P	C
1	PCP1226	Parallel Programming	PE	3	3	0	0	3
2	PCP1227	Software Architecture and Design	PE	3	3	0	0	3
3	PCP1279	Image Processing and Analysis	PE	3	3	0	0	3
4	PCP1228	Speech Processing and Synthesis	PE	3	3	0	0	3
5	PCP1229	Bio Inspired Computing	PE	3	3	0	0	3
6	PCP1231	Bio Informatics	PE	3	3	0	0	3
7	PCP1232	Advanced Algorithms	PE	3	3	0	0	3
8	*********	Network Security	PE	3	3	0	0	3

Professional Elective 3

Sl.No	Course Code	Course Title	Category	Contact Periods	L	T	P	C
1	PCP1376	Real Time Systems	PE	3	3	0	0	3
2	PCP1321	Software Project Management	PE	3	3	0	0	3

Sl.No	Course Code	Course Title	Category	Contact Periods	L	T	P	C
3	PCP1322	Computer Vision	PE	3	3	0	0	3
4	PCP1377	Information Retrieval Techniques	PE	3	3	0	0	3
5	PCP1378	Data Visualization Techniques	PE	3	3	0	0	3
6	PCP1323	Software Defined Networks	PE	3	3	0	0	3
7	PCP1324	Logic in Computer Science	PE	3	3	0	0	3

Professional Elective 4

Sl.No	Course Code	Course Title	Category	Contact Periods	L	T	P	C
1	PCP1325	Mobile and Pervasive Computing	PE	3	3	0	0	3
2	PIF1176	Agile Software Engineering	PE	3	3	0	0	3
3	PCP1326	Social Network Analysis	PE	3	3	0	0	3
4	PCP1379	Deep Learning	PE	3	3	0	0	3
5	PCP1327	Formal Verification	PE	3	3	0	0	3
6	PCP1328	Storage and Server Security	PE	3	3	0	0	3

Open Electives (OE)

Sl.No	Course Code	Course Title	Category	Contact Periods	L	T	P	C
1		Scientific Programming Using Python	OE	4	2	0	2	3
2		Introduction to Machine Learning	OE	4	2	0	2	3
3		Introduction to Big Data Analytics	OE	4	2	0	2	3

Areas of Specialization

	Sem 2		Sem 3	
	PE 1	**PE 2**	**PE 3**	**PE 4**
Distributed Computing	Cloud Computing, Advanced Databases (CN)	Parallel Programming	Real Time Systems	Mobile and Pervasive Computing
Software Engineering	Web Engineering	Software Architecture and Design	Software Project Management	Agile Software Engineering
Data Science	Natural Language Processing	Image Processing and Analysis, Speech Processing and Synthesis, Bio Inspired Computing, Bio Informatics	Computer Vision, Information Retrieval, Data Visualization Techniques	Social Network Analysis, Deep Learning
Computer Networks	Wireless Networks		Software Defined Networks	
Theoretical Computer Science	Principles of Programming Languages	Advanced Algorithms (DC)	Logic in Computer Science	Formal Verification
Security	Introduction to Cryptography	Network Security (CN)		Storage and Server Security (DC, CN)

SUMMARY

S. NO.	SUBJECT AREA	CREDITS AS PER SEMESTER				CREDITS TOTAL	PERCENTAGE
		I	II	III	IV		
1	FC	4				4	**5.5**
2	PC	18	15			33	**45.20**
3	EEC	1	2	6	12	21	**28.76**
4	OE			3		3	**4.1**
5	PE		6	6		12	**16.44**
		23	**23**	**15**	**12**	**73**	**100**

Course Code	Course Title	Category	L	T	P	C
PMA1177	APPLIED PROBABILITY AND STATISTICS	FC	4	0	0	4

OBJECTIVES

- To provide a foundation on topics in applied probability and statistical methods needed for modern optimization methods and risk modelling.
- To address the issues and the principles of estimation theory, testing of hypothesis and multivariate analysis.

UNIT I PROBABILITY AND RANDOM VARIABLES 12

Random variables – Probability function – Moments – Moment generating functions and their properties – Binomial, Poisson, Geometric, Uniform, Exponential, Gamma and Normal distributions.

UNIT II TWO DIMENSIONAL RANDOM VARIABLES 12

Joint distributions – Marginal and conditional distributions – Transformation of two-dimensional random variables – Correlation and regression

UNIT III ESTIMATION THEORY 12

Unbiased estimators – Method of moments – Maximum likelihood estimation – Curve fitting by principle of least squares – Regression lines.

UNIT IV TESTING OF HYPOTHESIS 12

Large sample test based on Normal distribution for single mean and difference of means – Tests based on t, and F distributions for testing means and variances – Contingency table (Test for Independency) – Goodness of fit.

UNIT V MULTIVARIATE ANALYSIS 12

Random vectors and matrices – Mean vectors and covariance matrices – Multivariate normal density and its properties – Principal components – Population principal components – Principal components from standardized variables

TOTAL PERIODS: 60

OUTCOMES

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

- To identify the standard distributions and apply them in solving problems(K3).
- To understand the concept of two-dimensional random variables and solve problems in finding the Joint probabilities and correlation between them(K3).
- Evaluate the consistency, efficiency and unbiasedness of estimators and apply for problems in data science (K3).
- Apply statistical tests in testing hypotheses on data (K3).

- Develop exploratory analysis of multivariate data, such as multivariate normal density and Population principal components. (K3).

REFERENCE BOOKS

1. Devore J. L., Probability and Statistics for Engineering and the Sciences, Cengage Learning, 8th Edition, 2014.
2. Dallas E. Johnson, Applied Multivariate Methods for Data Analysis, Thomson and Duxbury Press, 1998.
3. Gupta S. C., Kapoor V. K., Fundamentals of Mathematical Statistics, Sultan and Sons, New Delhi, 2001.
4. Johnson R. A., Miller I. and Freund J., Miller, Freund's Probability and Statistics for Engineers, Pearson Education, Asia, 8th Edition, 2015.
5. Richard A. Johnson, Dean W. Wichern, Applied Multivariate Statistical Analysis, Pearson Education, Asia, 5th Edition, 2002.

Course Code	Course Title	Category	L	T	P	C
PCP1176	ADVANCED DATA STRUCTURES AND ALGORITHMS	PC	3	0	0	3

OBJECTIVES

- To understand the use of algorithms in computing.
- To learn develop algorithms using hierarchical data structures.
- To select and use data structures and algorithms appropriate to the problems.
- To study about NP hard and NP completeness of problems.

UNIT I ROLE OF ALGORITHMS IN COMPUTING 9

Algorithms – Analyzing algorithms – Designing algorithms – Growth of functions: Asymptotic notation – Standard notations and common functions; Recurrences: The substitution method – The recursion-tree method – The master method for solving recurrences.

UNIT II HIERARCHICAL DATA STRUCTURES 9

Binary search trees: Basics – Query, Insertion and Deletion; Red-black trees: Properties of Red-black trees – Rotations – Insertion – Deletion; B-Trees: Basic operations on B-Trees; Fibonacci heaps: Structure – Mergeable-heap operations – Decreasing a key and deleting a node; Disjoint-set operations – Disjoint-set forests.

UNIT III GRAPHS 9

Elementary graph algorithms: Representations of graphs – Breadth-first search – Depth-first search – Topological sort – Strongly connected components; Single source shortest paths: Bellman-Ford algorithm; Single-source shortest paths in directed acyclic graphs: Dijkstra’s algorithm; All-pairs shortest paths: Floyd Warshall algorithm

UNIT IV ALGORITHM DESIGN TECHNIQUES 10

Greedy Algorithms: The job/event scheduling problem – Minimum-spanning tree problem; Recursive backtracking: Developing recursive backtracking algorithm – Pruning branches – N-queens problem; Developing dynamic programming algorithms – Subtle points – Decreasing time and space – Longest common subsequence problem, Weighted job/Event Scheduling Problem

UNIT V NP COMPLETE AND NP HARD 8

Reductions and NP – Completeness – Satisfiability – Proving NP-Completeness – 3-Coloring – Bipartite Matching.

TOTAL PERIODS: 45

OUTCOMES

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

- Apply asymptotic notations and methods to find the time complexity of iterative and recursive algorithms. (K3)
- Develop optimal algorithms using tree data structures (K3)
- Develop optimal algorithms using graph data structures. (K3)
- Solve problems using suitable design techniques (K3)

- Explain NP type problems and their reductions (K2)

REFERENCE BOOKS

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, Introduction to Algorithms, Prentice-Hall, Third Edition, 2012.
2. Jeff Edmonds, How to Think about Algorithms, Cambridge University Press, 2008.
3. Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, Data Structures and Algorithms, Pearson Education, Reprint 2006.
4. Robert Sedgewick, Kevin Wayne, Algorithms, Pearson Education, Fourth Edition, 2012.
5. Sridhar S., Design and Analysis of Algorithms, Oxford University Press, First Edition, 2014.

Course Code	Course Title	Category	L	T	P	C
PCP1101	**MULTICORE ARCHITECTURE AND GPU COMPUTING**	**PC**	**3**	**0**	**0**	**3**

OBJECTIVES

- To understand the basics of multi-core architectures.
- Understand the basics of SIMD and MIMD systems.
- Learn about CUDA programming constructs, and solutions.

UNIT I MULTICORE PROCESSORS 9

Single core to multicore architectures; Homogeneous and heterogeneous Multicore architectures: Intel multicore architectures – SUN CMP architecture – IBM cell architecture; Introduction to warehouse-scale computers architectures – Physical infrastructure and costs.

UNIT II MULTI PROCESSOR ARCHITECTURES AND ISSUES 9

MIMD systems: Symmetric and distributed shared memory architectures; Cache coherence; Interconnection networks; Performance issues; Parallel program design.

UNIT III VECTOR, SIMD AND GPU ARCHITECTURES 9

SIMD Systems: Introduction – Vector architecture – SIMD extensions for multimedia – Graphics Processing Units; Detecting and enhancing loop level parallelism; Case studies – GPGPU computing.

UNIT IV GPU PROGRAMMING USING CUDA 10

CUDA Hardware Overview: Threads – Blocks – Grids – Warps – Scheduling; Memory handling with CUDA: Shared memory – Global memory – Constant memory and texture memory. Common Problems: CUDA error handling – Parallel programming issues – Synchronization – Algorithmic issues – Finding and avoiding Errors.

TOTAL PERIODS: 45

OUTCOMES

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

- Infer the salient features of different multicore architectures and how they exploit parallelism. (K2)
- Interpret the features and issues of MIMD systems. (K2)
- Explain the features of SIMD systems. (K2)
- Develop programs using CUDA. (K3)
- Apply and identify issues in CUDA programming. (K3)

REFERENCE BOOKS

1. John L. Hennessey, David A. Patterson, Architecture – A Quantitative Approach, Morgan Kaufmann / Elsevier, 5th edition, 2012. (units I, III)
2. Peter S. Pacheco, An Introduction to Parallel Programming, Morgan Kaufmann, 2014. (unit II)
3. Shane Cook, CUDA Programming: A Developer’s Guide to Parallel Computing with GPUs (Applications of GPU Computing), Morgan Kaufmann, First Edition, 2012. (unit IV, V)

4. Sanders J., Kandrot E. CUDA by example: An Introduction to General Purpose GPU Programming, Addison-Wesley, 2011.
5. Nicholas Wilt, CUDA Handbook: A Comprehensive Guide to GPU Programming, Addison - Wesley, 2013.

Course Code	Course Title	Category	L	T	P	C
PCP1177	ADVANCED SOFTWARE ENGINEERING	PC	3	0	0	3

OBJECTIVES

- To understand requirements specification and apply UML diagrams.
- To understand and apply software architectural styles and design patterns.
- To understand and apply software testing techniques.
- To be familiar with Agile software development methodology.
- To understand DevOps practices

UNIT I SOFTWARE REQUIREMENTS SPECIFICATION 9

Requirement analysis and specification – Requirements gathering and analysis – Formal system specification – Finite State Machines – Petrinets – Object modelling using UML – Use case Model – Class diagrams – Interaction diagrams – Activity diagrams – State chart diagrams – Functional modelling – Data Flow Diagrams.

UNIT II SOFTWARE ARCHITECTURE AND DESIGN 9

Architectural styles – Layered – Client/server – Tiered – Pipe and filter- Software design – Design process – Design concepts – Coupling – Cohesion – Functional independence –Design patterns – Model-view-controller – Publish-subscribe – Adapter – Command – Strategy – Observer – Proxy – Facade - User interface design.

UNIT III SOFTWARE TESTING 9

Testing – Unit testing – Black box testing– White box testing – Integration and System testing– Regression testing – Debugging – Program analysis – Symbolic execution – Model Checking.

UNIT IV AGILE SOFTWARE DEVELOPMENT 9

Introduction to Agile Methodology – Evolution of Agile – Self adaption: Growing and Reflection – Pair Programming – Agile methods – collaboration – Releasing – Planning – Development.

UNIT V DEVOPS 9

DevOps: Motivation-Cloud as a platform – Operations – Deployment Pipeline: Overall Architecture – Building and Testing – Deployment – Cross cutting Concerns: Monitoring – Security Audits Deployment – Case study: Migrating to Micro services.

TOTAL PERIODS: 45

OUTCOMES

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

- Explain software requirements specification and model software systems using UML diagrams (K3)
- Apply formal models to capture requirements (K3)
- Apply appropriate architectural styles and design patterns (K3)
- Apply various testing techniques (K3)
- Describe agile software development methodology (K2)
- Summarize the benefits of DevOps practices (K2)

REFERENCE BOOKS

1. Rajib Mall, Fundamentals of Software Engineering, PHI Learning Pvt Ltd, 3rd edition, 2009. (Unit I, II & III)
2. Carlo Ghezzi, Mehdi Jazayeri, Dino Mandrioli, Fundamentals of Software Engineering, PHI Learning Pvt Ltd, 2nd edition, 2010. (Unit I & III)
3. Mary Shaw, David Garlan, Software Architecture: Perspectives on an Emerging Discipline, Pearson Education, 1996. (Unit II)
4. Craig Larman, Applying UML and Patterns, Pearson Education, 3rd edition, 2005. (Unit I, II & III)
5. Alistair Cockburn, Jim Highsmith, Agile Software Development – The Cooperative Game, Pearson Education, Second edition, 2007.
6. James Shore, Shane Warden, The Art of Agile Development, O'Reilly, Shroff Publishers 2011. (Unit IV)
7. Len Bass, Ingo Weber, Liming Zhu, DevOps: A Software Architect's Perspective, Pearson Education, 2016. (Unit V)

Course Code	Course Title	Category	L	T	P	C
PCP1178	MACHINE LEARNING	PC	3	0	2	4

OBJECTIVES

- To have a basic knowledge of the concepts and techniques of machine learning.
- To understand the working of various machine learning algorithms.
- To use the various probability-based learning techniques and evolutionary
- models.
- To understand graphical models.

UNIT I INTRODUCTION 8

Learning: Types of machine learning – Design of a learning system – Perspectives and issues in machine learning; Concept Learning Task: Concept learning as search – Finding a maximally specific hypothesis – Version spaces and Candidate elimination algorithm; Curse of dimensionality – Overfitting – Bias variance tradeoff.

UNIT II LINEAR AND NON-LINEAR MODELS 10

The Brain and the Neuron – Perceptron – Linear separability – Linear regression; Multi-Layer Perceptron: Going forwards – Going backwards – Back propagation error – Multi-layer perceptron in Practice – Examples of using the MLP – Deriving back-propagation; Radial Basis Functions and Splines: Concepts – RBF Network; Support Vector Machines: Kernels.

UNIT III TREE AND PROBABILISTIC MODELS 9

Learning with Trees: Decision trees – Constructing decision trees – Classification and regression trees; Ensemble Learning: Boosting – Bagging – Different ways to Combine Classifiers; Probabilistic Learning: Gaussian Mixture Models – Nearest neighbour methods; Unsupervised Learning: K-means algorithms.

UNIT IV DIMENSIONALITY REDUCTION AND EVOLUTIONARY MODELS 9

Dimensionality Reduction: Linear discriminant analysis – Principal component analysis – Independent component analysis; Evolutionary Learning: Genetic algorithms – Genetic offspring – Genetic operators – Using Genetic algorithms; Reinforcement Learning: ‘Getting lost’ example – Markov decision process.

UNIT V GRAPHICAL MODELS 9

Markov Chain Monte Carlo Methods: Sampling – Proposal distribution – Markov Chain Monte Carlo; Graphical Models: Bayesian networks – Markov Random Fields – Hidden markov models

Suggestive Experiments (Python – Numpy, Scipy, Scikit-learn, Matplotlib)

- Perceptron and Linear Regression
- Multi-layer Perceptron
- Support Vector Machine
- Decision Tree algorithm
- k-Nearest Neighbor algorithm

- K-means clustering
- Random Forest and AdaBoost ensemble techniques
- Dimensionality reduction techniques : LDA, PCA

TOTAL PERIODS: 75

OUTCOMES

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

- Explain the basic concepts of machine learning (K2)
- Analyze linear and non-linear techniques for classification problems (K4)
- Apply tree and probabilistic models for the given problems (K3)
- Apply various dimensionality reduction techniques and evolutionary models (K3)
- Explain the concepts of graphical models (K2)

REFERENCE BOOKS

1. Stephen Marsland, Machine Learning – An Algorithmic Perspective, Chapman and Hall/CRC Machine Learning and Pattern Recognition Series, Second Edition, 2014.
2. Tom M Mitchell, Machine Learning, McGraw Hill Education, First Edition, 2013.
3. Ethem Alpaydin, Introduction to Machine Learning 3e (Adaptive Computation and Machine Learning Series), MIT Press, Third Edition, 2014.
4. Jason Bell, Machine learning – Hands on for Developers and Technical Professionals, Wiley, First Edition, 2014
5. Peter Flach, Machine Learning: The Art and Science of Algorithms that Make Sense of Data, Cambridge University Press, First Edition, 2012.

Course Code	Course Title	Category	L	T	P	C
PCP1102	NETWORK DESIGN AND TECHNOLOGIES	PC	2	0	2	3

OBJECTIVES

- To understand the principles of network design.
- To explore various technologies in the wireless domain.
- To study about 3G, 4G and cellular networks.
- To understand the paradigm of software defined networks.
- To design networks using simulator.

UNIT I NETWORK DESIGN 6

OSI Model – Review of Protocols – LAN cabling topologies – Ethernet switches, routers, firewalls and L3 switches – Remote access technologies and devices – Core networks and distribution networks.

UNIT II WIRELESS NETWORKS 6

IEEE802.16 and WiMAX – Security – Advanced 802.16 functionalities – Mobile WiMAX:802.16e – Network infrastructure – WLAN – Configuration – Management operation.

UNIT III CELLULAR NETWORKS 6

GSM – Mobility management and call control – The Mobile Device – The SIM card – GPRS – Network elements – Mobility management and session management.

UNIT IV 4G NETWORKS 6

LTE – Network architecture and interfaces – Mobility management and power optimization – LTE security architecture – Interconnection with UMTS and GSM – 4G networks and composite radio environment – Protocol boosters – Hybrid 4G wireless networks protocols.

UNIT V SOFTWARE DEFINED NETWORKS 6

Introduction – Centralized and distributed control and data planes –Open flow – Wire protocol – Architecture – SDN controller framework.

Suggestive Exercises

- Establishing LAN
- Creation of Virtual LAN
- Simulation of Wired networks
- Simulation of Wireless networks with access points
- Analyzing the packet flow using Wireshark Tool
- Simulation of Routing Protocol RIP
- Simulation of Routing Protocol OSPF
- Simulation of interdomain redistribution of RIP- OSPF
- Creation of Access Control List in the server firewall

TOTAL PERIODS: 60

OUTCOMES

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

- Outline the components required for designing a network (K2).
- Develop a network using different wireless networking technologies (K3).
- Experiment with various protocols of 3G, 4G and cellular networks (K3).
- Make use of software defined networks (K3).
- Analyze various networks using a simulator (K4).

REFERENCE BOOKS

1. Larry Peterson, Bruce Davie, Computer Networks: A Systems Approach, Morgan Kauffman, 5th edition, 2011.
2. Martin Sauter, From GSM to LTE, An Introduction to Mobile Networks and Mobile Broadband, Wiley, 2014.
3. Savo G Glisic, Advanced Wireless Networks – 4G Technologies, John Wiley & Sons, 2007.
4. Thomas D. Nadeau and Ken Gray, SDN – Software Defined Networks, O'Reilly Publishers, 2013.

Course Code	Course Title	Category	L	T	P	C
PCP1197	ADVANCED DATA STRUCTURES AND ALGORITHMS LAB	PC	0	0	4	2

OBJECTIVES

- To learn advanced tree and graph structures.
- To learn the need for heap and disjoint set representations.
- To understand algorithm design techniques and apply suitable techniques for problems.

Suggestive Programming Languages

- C, Java or Python

Suggestive Exercises

- Insertion sort and merge sort with complexity analysis
- Application of binary search trees
- Red-Black Trees
- Min heaps, Fibonacci heaps
- Disjoint sets
- Application of graph traversals – BFS and DFS
- Single source Shortest Path Algorithms (e.g. Dijkstra’s and Bellman Ford algorithms)
- All-pairs Shortest Path Algorithms: (e.g. Floyd’s algorithm)
- Dynamic programming (e.g. Longest common subsequence, 0/1 Knapsack)
- Recursive backtracking (e.g. N-Queen’s problem)
- Greedy technique (e.g. Job/event scheduling, Minimum spanning tree)
- Mini project: For a given application, identify and solve using suitable data structures and design techniques.

TOTAL PERIODS: 60

OUTCOMES

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

- Solve problems using basic data structures and sorting algorithms. (K3)
- Solve problems using tree and graph structures. (K3)
- Analyse algorithm design techniques and develop efficient algorithms (K4)

Course Code	Course Title	Category	L	T	P	C
PCP1111	ETHICAL PRACTICES	EEC	0	0	2	1

OBJECTIVES

- To create an awareness on Engineering Ethics and Human Values.
- To understand social responsibility of an engineer.
- To appreciate ethical dilemma while discharging duties in professional life.

Topics

Engineering Ethics (8 hours)
Responsibility in Engineering, Analysing interior and exterior intensions, Moral judgements and responsibility

Human Values (4 hours)
Morals, Values and Ethics – Integrity – Work Ethic – Honesty – Courage – Empathy – Self-Confidence – Character

Social and Value Dimensions in Technology (6 hours)
Privacy and social policy, ethical issues in design, Approaches to risk, case studies

Engineers in Organizations (6 hours)
Organizational culture, Clean environment, Fairness: person to person and social

Best Practices (6 hours)
Moral behaviour, Best practices in engineering, Case studies

TOTAL PERIODS: 30

OUTCOMES

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

- Analyse human values and virtues (K4).
- Analyse the importance of ethical practices in engineering (K4).
- Apply ethical practices in engineering and technology (K3).
- Make use of the organizational environment and culture (K3).
- Apply best practices in engineering and analyse moral behaviour & responsibility (K4).

REFERENCE BOOKS

- Charles E Harris, Michael S. Protchard, Michael J Rabins, Engineering Ethics –Concepts and Cases, Wadsworth Thompson Learning, United States, 2000 (Indian Reprint now available).
- Edmund G Seebauer, Robert L Barry, Fundamentals of Ethics for Scientists and Engineers, Oxford University Press, Oxford, 2001.
- Charles D. Fleddermann, Engineering Ethics, Pearson Education/Prentice Hall, New Jersey, 2004 (Indian Reprint now available).
- John R Boatright, Ethics and the Conduct of Business, Pearson Education, New Delhi, 2003.

Course Code	Course Title	Category	L	T	P	C
PCP1201	MODERN OPERATING SYSTEMS	PC	3	0	2	4

OBJECTIVES

- To understand the design of operating systems.
- To learn about the contemporary operating systems.
- To explore the design of mobile operating systems.
- To apply the knowledge of scheduling in real-time problems.
- To apply the knowledge of resource handling techniques on real-time problems.

UNIT I VIRTUALIZATION AND MULTIPROCESSOR SYSTEMS 9

History – Requirements – 1 and Type 2 Hypervisors – Techniques for Efficient Virtualization – Microkernels – Memory and I/O Virtualizations – Clouds – Case Study: Vmware.

UNIT II DESIGN OF MOBILE OS 9

Android and Google – History of Android – Design Goals – Architecture – Linux Extensions – Dalivik – Binder IPC – Android Applications – Intents – Application Sandboxes– Security – Process Model – Typical System Architecture – SoC – Memory layout and Mapping – Development setup – Evaluation Boards.

UNIT III SCHEDULING IN REAL-TIME SYSTEMS 9

Introduction – Applications – BasicModel – Characteristics of Real time system – Safety – Task types – Timing Constraints – Concepts and Characteristics of types of Real Time Task Scheduling – Clock Driven Scheduling – Event Driven Scheduling – Earliest Deadline First – Rate Monotonic Algorithm.

UNIT IV RESOURCE HANDLING TECHNIQUES IN REAL-TIME SYSTEMS 9

Resource Sharing – Priority Inversion – Priority Inheritance Protocol – Highest Locker Protocol – Priority Ceiling Protocol– Types of Priority Inversions – Important Features – Issues and Task dependencies.

UNIT V OPERATING SYSTEM DESIGN 9

The Nature of the design Problem – Interface Design – Implementation– Performance – Project Management – Trends in Operating System Design.

Suggestive Exercises

- Installation of VMware hypervisor
- Performing Migrations between VMs
- Implementation of Rata Monotonic Algorithm
- Implementation of Cyclic Scheduler
- Implementation of Priority Ceiling Protocol

TOTAL PERIODS: 75

OUTCOMES

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

- Apply various techniques related to virtualization (K3)
- Describe the build of Android operating system (K2)
- Apply scheduling algorithms in real-time applications (K3)
- Apply resource handling techniques in real-time environments (K3)
- Install and work with modern OS and comprehend its practical design (K3)

REFERENCE BOOKS

1. Andrew S. Tanenbaum, Herbert Bos, Modern Operating Systems, Pearson, 4th Edition, 2015 (Units 1, 2 and 5)
2. Rajib Mall, Real Time Systems: Theory and Practice, Pearson, 2007 (Units 4 and 5).
3. Karim Yaghmour, Embedded Android, O-Rielly, 2013 (Unit 2).
4. Matthew Portnoy, Virtualization Essentials, Wiley, 2012.
5. Jerome Saltzer M., Frans Kaashoek, Principles of Computer System Design, Elsevier, 1st Edition, 2009.
6. Joshua Drake et al, Android Hackers Handbook, Wiley, 2014.
7. Francis Cottet, Joelle Delacroix, et al, Scheduling in Real-Time Systems, Wiley, 2002.

Course Code	Course Title	Category	L	T	P	C
PCP1276	**INTERNET OF THINGS**	**PC**	**2**	**0**	**2**	**3**

OBJECTIVES

- To understand the fundamentals of Internet of Things.
- To learn about the basics of IOT protocols.
- To build a small low-cost embedded system using Arduino/Raspberry Pi.
- To apply the concept of Internet of Things in real world scenarios.

UNIT I INTRODUCTION 6

M2M and IoT Technology Fundamentals – Devices and gateways, Local and wide area networking, Data management, Business processes in IoT, Everything as a Service (XaaS), M2M and IoT Analytics, Knowledge Management. IoT-An Architectural Overview – Building an architecture, An IoT architecture outline.

UNIT II IoT REFERENCE ARCHITECTURE 6

IoT Architecture: State of the Art – Reference Model: Domain model – information model – functional model – communication model – IoT Reference Architecture: Functional View – Information View.

UNIT III IoT PROTOCOLS 6

Protocol Standardization for IoT – Efforts - M2M andWSN Protocols - SCADA and RFID Protocols - Unified Data Standards - Protocols - IEEE 802.15.4 - Zigbee Architecture - Network layer – 6LowPAN – CoAP.

UNIT IV BUILDING IoT SYSTEMS 6

Logical design using Python – IoT Physical devices and Endpoints: Basic building blocks of IoT Device – Raspberry Pi – Linux on Raspberry Pi – Interfaces – Programming Raspberry Pi with Python – Other IoT Platform: Arduino.

UNIT V CASE STUDIES AND REAL WORLD APPLICATIONS 6

Home Automation – Weather monitoring – Smart irrigation – Data Analytics for IoT – Software & Management Tools for IoT Cloud Storage Models & Communication APIs – Cloud for IoT.

Suggestive Exercises (using Arduino / Raspberry Pi Board)

- Study of Arduino and Raspberry Pi Board
- LED blinking and fading
- Push button and LDR interfacing
- IR and buzzer interfacing
- DC motor interfacing
- Joystick interfacing
- Stepper motor interfacing
- PIR motion sensor interfacing
- IMU interfacing

- Temperature and humidity sensor interfacing
- Application development using Arduino / Raspberry Pi using various sensors (Examples: Smart parking, Smart traffic control)
- Application development by integrating Arduino and Raspberry Pi in cloud platform (Examples: Basic home automation, Weather monitoring)

TOTAL PERIODS: 60

OUTCOMES

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

- Explain the basic building blocks and data management in Internet of Things (K2).
- Explain different IoT architectures (K2).
- Summarize various protocols for IoT (K2).
- Develop a low cost IoT system using Arduino/Raspberry-Pi (K3).
- Build an IoT application in a cloud environment (K3).

REFERENCE BOOKS

1. Jan Holler, Vlasios Tsiatsis, et al, From Machine-to-Machine to the Internet of Things – Introduction to a New Age of Intelligence, Elsevier, 2014.
2. Honbo Zhou, The Internet of Things in the Cloud: A Middleware Perspective, CRC Press, 2012.
3. Olivier Hersent, David Boswarthick, Omar Elloumi, The Internet of Things – Key applications and Protocols, Wiley, 2012.
4. Arshdeep Bahga, Vijay Madisetti, Internet of Things – A Hands-on Approach, Universities Press, 2015.

Course Code	Course Title	Category	L	T	P	C
PCP1202	WEB APPLICATION DEVELOPMENT	PC	3	0	0	3

OBJECTIVES

- To understand the concepts of web site development
- To learn web application development using JavaScript based framework.
- To learn web application development using Python based framework.

UNIT I INTRODUCTION 9

Web Essentials: Web clients, web servers, communication protocol, HTTP request and response message; HTML5: Tables - Lists - Image – Control and semantic elements, audio and video controls –CSS3 - Inline, embedded and external style sheets – Rule cascading – Inheritance – Backgrounds – Border Images – Colors – Shadows – Text – Transformations – Transitions – Animations.

UNIT II CLIENT SIDE PROGRAMMING 9

Introduction to JavaScript – Variables – Data types – Statements – Operators – Functions –Objects – Arrays – Built-in objects – Event handling – Introduction to DOM – The document tree – Node object – methods and properties – Document object – Methods and properties – Addition and deletion of elements – Modifying element style.

UNIT III JAVASCRIPT BASED FRAMEWORK 9

Introduction to AngularJS – Anatomy of AngularJS Application – Data binding – Model – Module – View – Controller – Expressions – Scope – Table – Select – Form – Validation – DOM – Event – Directives.

UNIT IV WEB APPLCIATION USING DJANGO 9

Building a blog: Creating a project – Running the development server – Designing the model – Setting up the database; Dynamic website basics – MVC – Django architecture - Defining and using the models: ORM – URL configurations – Modelling HTTP – Views/Logics - Templates and form processing.

UNIT V WEB APPLCIATION USING ADVANCED DJANGO 9

Django applications: Photo gallery – Content management system – Liveblog; Advanced Django programming: Customizing the admin – Syndication – Generating downloadable files – Enhancing Django’s ORM – Extending the template system; Advanced Django deployment.

TOTAL PERIODS: 45

OUTCOMES

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

- Create a website using HTML and CSS (K3).
- Develop web applications using the JavaScript-based framework (K3).
- Develop web applications using a Python-based framework (K3).

REFERENCE BOOKS

1. Jeffrey C Jackson, Web Technologies A Computer Science Perspective, Pearson Education, 2011. (unit I)
2. Deitel and Deitel, Nieto, Internet and World Wide Web - How to Program, Prentice Hall, 5th Edition, 2011. (unit II)
3. Brad Green, Shyam Seshadri, AngularJS, O'Reilly Media, First Edition, 2013. (unit III)
4. Jeff forcier, Paul bissex, Wesley chun, Python web development with Django, Addison-Wesley Professional, 2008. (unit IV, V)

Course Code	Course Title	Category	L	T	P	C
PCP1281	DATA ANALYTICS	PC	3	0	0	3

OBJECTIVES

- To understand the competitive advantages of data analytics.
- To understand data frameworks.
- To learn data analysis methods.
- To learn stream computing.
- To gain knowledge on Hadoop related tools such as HBase, and Hive for data analytics.

UNIT I INTRODUCTION TO BIG DATA 8

Big Data: Definition – Characteristic features – Big data applications – Big data vs traditional data – Risks of big data – Structure of big data; Web data; Evolution of analytic scalability; Modern Data Analytic Tools: R programming.

UNIT II HADOOP FRAMEWORK 9

Distributed File Systems: Large-scale file system organization – HDFS concepts – MapReduce execution – Algorithms using MapReduce.

UNIT III DATA ANALYSIS 11

Statistical Methods: Regression modelling – Multivariate analysis; Classification: SVM & Kernel methods; Cluster Analysis: Types of data in cluster analysis – Partitioning methods – Hierarchical methods – Density based methods – Model based clustering methods – Clustering high dimensional data.

UNIT IV MINING DATA STREAMS 8

Streams: Concepts – Stream data model and architecture – Sampling data in a stream; Mining Data Streams: 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.

UNIT V BIG DATA FRAMEWORKS 9

Introduction to NoSQL – MongoDB – Aggregate data models – Hbase: Data model and implementations – Hbase clients; Hive: Data types and file formats – HiveQL data definition – Hive QL data manipulation – HiveQL queries

TOTAL PERIODS: 45

OUTCOMES

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

- Explain how to leverage the insights from big data analytics (K2).
- Solve applications using statistical and data analytic methods (K3).
- Develop analytics using streaming data (K3).
- Develop applications using Hadoop-related tools and R Programming (K4).
- Develop databases using MongoDB, Hive, and HBase for data analysis (K3).

REFERENCE BOOKS

1. Bill Franks, Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics, John Wiley & Sons, 2012.
2. Anand Rajaraman, Jeffrey David Ullman, Mining of Massive Datasets, Cambridge University Press, 2012.
3. Roger D. Peng, R programming for Data Science, LeanPub, 5th edition, 2016.
4. Michael Berthold, David J. Hand, Intelligent Data Analysis, Springer, 2007.
5. Tom White, Hadoop: The Definitive Guide – Storage and Analysis at Internet Scale, O'Reilly, 4th Edition, 2015.
6. Capriolo E., Wampler D., Rutherglen J., Programming Hive, O'Reilly, 2012.
7. Lars George, HBase: The Definitive Guide, O'Reilly, 2011.
8. Sadalage P. J., Fowler M., NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence, Addison - Wesley Professional, 2012.
9. Kristina Chodorow, MongoDB: The Definitive Guide – Powerful and Scalable Data Storage, O'Reilly, 2nd Edition, 2013.

Course Code	Course Title	Category	L	T	P	C
PCP1211	**DATA ANALYTICS LAB**	**PC**	**0**	**0**	**4**	**2**

OBJECTIVES

- To implement Map-Reduce programs for processing big data.
- To realize storage of big data using HBase, MongoDB.
- To analyse big data using linear models.
- To analyse big data using machine learning techniques for classification and clustering.

Suggestive Experiments

Hadoop

- Applications using Map-Reduce programming (Examples: word count / frequency programs / matrix multiplication)
- Implement an MR program that processes a dataset.

R

- Linear and logistic Regression (Loan prediction using Credit approval dataset, Sales prediction using Bigmart dataset)
- SVM /Decision tree classification techniques (Flower type classification based on available attributes using Iris dataset, Passengers survival classification using titanic dataset)
- Multilayer perceptron (Handwritten digit classification using neural network)
- Clustering (Document categorization by multiclass techniques)
- Visualize data using any plotting framework.

Database

- Application that stores data in HBase / MongoDB (Sentiment analysis using twitter dataset)

TOTAL PERIODS: 60

OUTCOMES

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

- Solve big data analysis using Hadoop framework (K3).
- Build and apply linear and logistic regression models (K3)
- Examine data analysis with machine learning methods (K4)
- Analyse data using visualization techniques (K4).

Course Code	Course Title	Category	L	T	P	C
PCP1217	**MINI PROJECT**	**EEC**	**0**	**0**	**2**	**1**

OBJECTIVES

- To formulate a problem and a solution,
- To solve a given problem using latest technologies.

Suggestive Mini projects

- Create a college website using HTML5 and CSS3
- Develop a MVC based web application for a Super Market Billing System. Use cookies as part of the web application.
- Develop online exam web application using AngularJS.
- Develop Library Management System using MVC Django Architecture
- Develop Inventory Management System using Django Framework

TOTAL PERIODS: 30

OUTCOMES

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

- Analyse the problem and design a solution (K4).
- Analyse and adopt new technologies for solving a given problem (K4).
- Solve problems using software engineering principles (K3).

Course Code	Course Title	Category	L	T	P	C
PCP1215	TECHNICAL PAPER WRITING AND SEMINAR	EEC	0	0	2	1

OBJECTIVES

- To develop scientific and technical reading and writing skills
- To understand and construct research articles.

A technical paper or report requires a student to obtain information from a variety of sources (i.e., Journals, dictionaries, reference books, conference proceedings, etc.), consolidate, draw conclusions and write as a report. The work involves the following steps:

- Selecting an area, narrowing the area into a topic
- Stating an objective
- Collecting the relevant literature
- Reading the papers, understanding the contributions of the authors, preparing an abstract or outline of the paper and discussing
- Critically analysing each paper that have been read, linking them, and drawing conclusions.
- Writing the technical report about the topic, literature, comparison, and observation
- Presenting the report

TOTAL PERIODS: 30

OUTCOMES

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

1. Construct technical reports clearly for a given topic (K3)
2. Analyze and select a topic, collect literature, consolidate (K4)
3. Develop an effective presentation of the literature survey and observations (K3)

COURSE CODE	COURSE TITLE	L	T	P	C
PCP1318	PROJECT WORK PHASE - I	0	0	12	6

OBJECTIVES

- To develop the ability to solve a specific problem right from literature.
- review till the successful solution of the same.
- To inculcate research culture
- To enhance the rational and innovative thinking capabilities
- To train the students in preparing project reports and to face reviews and viva voce examination.

COURSE OUTCOMES

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

- Formulate complex engineering problems and choose suitable technologies and tools required to solve them effectively (K5)
- Analyse the related literature on the proposed problem and its feasibility (K4)
- Design high level solutions by applying software engineering principles (K6)
- Implement a solution, evaluate, and deploy it. (K5)
- Document the solution, communicate orally and elaborate in writing.

COURSE CODE	COURSE TITLE	L	T	P	C
PCP1418	PROJECT WORK PHASE - II	0	0	24	12

OBJECTIVES

- To develop the ability to solve a specific problem right from literature.
- review till the successful solution of the same.
- To inculcate research culture
- To enhance the rational and innovative thinking capabilities
- To train the students in preparing project reports and to face reviews and viva voce examination.

COURSE OUTCOMES

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

- Plan the management of time and risks. (K3)
- Compare the related literature on the proposed problem and analyse its feasibility (K5)
- Design a high-level solution by applying software engineering principles (K6)
- Implement a solution, evaluate, and deploy it. (K5)
- Document the solution, communicate orally and elaborate in writing.

Course Code	Course Title	Category	L	T	P	C
PCP1277	CLOUD COMPUTING	PE	3	0	0	3

OBJECTIVES

- To understand the concepts of virtualization and virtual machines.
- To gain knowledge 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 CLOUD ARCHITECTURE MODELS AND INFRASTRUCTURE 9

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 – Global Exchange of Cloud Resources.

UNIT II 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 VIRTUALIZATION INFRASTRUCTURE 9

Comprehensive Analysis –Resource Pool – Testing Environment –Virtual Workloads – Provision 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 9

Google App Engine – Amazon AWS – Microsoft Azure; Cloud Software Environments – Eucalyptus – Open Stack – Open Nebula – Aneka – CloudSim.

UNIT V CLOUD SECURITY 9

Multi-tenancy Issues: Isolation of users/VMs from each other – Virtualization System Vulnerabilities: Management console vulnerabilities – management server vulnerabilities – administrative VM vulnerabilities – guest VM vulnerabilities, hypervisor vulnerabilities – hypervisor escape vulnerabilities – configuration issues; Virtualization System-Specific Attacks: Guest hopping – attacks on the VM (delete the VM, attack on the control of the VM, code or file injection into the virtualized file structure) – VM migration attack – hyperjacking.

TOTAL PERIODS: 45

OUTCOMES

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

- Describe the design challenges in cloud (K2)
- Apply the concept of virtualization and its types (K3)
- Experiment with virtualization of hardware resources and Docker (K3)
- Develop and deploy services on cloud and set up a cloud environment (K3)
- Explain security challenges in cloud environments (K2).

REFERENCE BOOKS

1. Danielle Ruest, Nelson Ruest, Virtualization: A Beginner’s Guide, McGrawHill Osborne Media, 2009. (Units I, II)
2. James E. Smith, Ravi Nair, Virtual Machines: Versatile Platforms for Systems and Processes, Elsevier/Morgan Kaufmann, 2005. (Units I, II)
3. William von Hagen, Professional Xen Virtualization, Wrox Publications, January 2008. (Unit II)
4. David Marshall, Wade A. Reynolds, Advanced Server Virtualization: VMware and Microsoft Platform in the Virtual Data Center, Auerbach Publications, 2006. (Unit II)
5. Srinivasan A, Suresh J, Cloud Computing: A practical Approach for Learning and Implementation, Pearson Education India, 2014. (Units I, II, III, IV)
6. Kai Hwang, Geoffrey C Fox, Jack G Dongarra, Distributed and Cloud Computing, From Parallel Processing to the Internet of Things, Morgan Kaufmann Publishers, 2012. (Units II, III)
7. Chris Hay, Brian Prince, Azure in Action, Manning Publications, 2010. (Unit IV)
8. Eugene Ciurana, Developing with GoogleApp Engine, Apress, 1 edition, 2009. (Unit IV)
9. Kevin Jackson, Cody Bunch, OpenStack Cloud computing Cookbook, Packt Publishing, Second Edition, 2013. (Unit IV)
10. Giovanni Toraldo, Open Nebula 3 Cloud Computing, Packt Publishing, 2012. (Unit IV)
11. Krutz, R. L., Vines, R. D, Cloud security. A Comprehensive Guide To Secure Cloud Computing, Wiley Publishing, 2010. (Unit V)
12. Tim Mather, Subra Kumaraswamy, Shahed Latif , Cloud Security and Privacy: an enterprise perspective on risks and compliance, O‘Reilly Media, Inc., 2009. (Unit V)

Course Code	Course Title	Category	L	T	P	C
PCP1221	ADVANCED DATABASES	PE	3	0	0	3

OBJECTIVES

- To acquire knowledge on parallel and distributed databases and its applications.
- To understand and apply the concepts of object and object-relational databases.
- To design and develop a web database using XML data model.
- To understand the emerging databases like mobile, and intelligent databases.

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

Object Database Concepts: Object-Oriented Concepts and Features – Object Identity, and Objects versus Literals – Complex Type Structures for Objects and Literals – Encapsulation of Operations and Persistence of Objects – Type Hierarchies and Inheritance – Other Object-Oriented Concepts; Object-Relational Features: Object Database Extensions to SQL; ODMG Object Model and ODL – Object Database Conceptual Design – Object Query Language OQL – C++ Language Binding in the ODMG Standard.

UNIT III XML AND WEB DATABASES 9

XML Databases: XML Data Model — DTD — XML Schema — XML Querying – XML Languages – Extracting XML Documents from Relational Databases; Introduction to Web Database.

UNIT IV MOBILE DATABASES 9

Mobile Databases: Location and Handoff Management; Effect of Mobility on Data Management: Data Categorization – Location Dependent Data Distribution; Mobile Transaction Models: HiCoMo – Moflex – Kangaroo – MDSTPM –Mobil action; Concurrency Control Mechanism – Transaction Commit Protocols – Mobile Database Recovery Schemes.

UNIT V INTELLIGENT DATABSES 9

Active Databases: Syntax and Semantics (Starburst, Oracle, DB2) – Taxonomy Applications – Design Principles for Active Rules; Temporal Databases: Overview of Temporal Databases TSQL2 – Deductive Databases–Recursive Queries in SQL; Spatial Databases: Spatial Data Types – Spatial Relationships– Spatial Data Structures–Spatial Access Methods– Spatial DB Implementation; Multimedia Database Concept

TOTAL PERIODS: 45

OUTCOMES

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

- Understand the parallel and distributed databases. (K3)
- Apply the object oriented database concepts and features. (K3)
- Describe and design XML databases. (K3)
- Understand the mobile databases and its transaction models. (K2)
- Understand the different types of intelligent databases. (K2)

REFERENCE BOOKS

1. Elmasri R., Navathe S. B., Fundamentals of Database Systems, Pearson Education/Addison Wesley, Sixth Edition, 2010. (Units I, II, III, V)
2. Vijay Kumar, Mobile Database Systems, John Wiley & Sons, 2006. (Unit IV)
3. Henry F Korth, Abraham Silberschatz, Sudharshan S., Database System Concepts, McGraw Hill, Sixth Edition, 2011.
4. Date C. J., Kannan A., Swamynathan S., An Introduction to Database Systems, Pearson Education, Eighth Edition, 2006.
5. Carlo Zaniolo, Stefano Ceri, Christos Faloutsos, Richard T. Snodgrass, Subrahmanian V. S., Roberto Zicari, Advanced Database Systems, Morgan Kaufmann publishers, 2006.

Course Code	Course Title	Category	L	T	P	C
PCP1222	WEB ENGINEERING	PE	3	0	0	3

OBJECTIVES

- Understand web engineering principles for web application development.
- Learn to architect and model web applications.
- Design web application using design methods.
- Learn various testing methods and understand web project management.

UNIT I INTRODUCTION TO WEB ENGINEERING 9

Web Engineering — Motivation -– Categories of Web Applications — Characteristics of Web Applications — Requirement Specifics in web engineering – Web engineering: Components; Process; Communication and Planning

UNIT II WEB APPLICATION ARCHITECTURE AND MODELLING WEB APPLICATIONS 9

Introduction – Categorizing architectures – Specifics of Web application architectures – Components of generic web application architecture -– Layered and Data-aspect architectures – Modelling specifics in web engineering —modelling: Requirements; Content; Hypertext; Presentation and Customization -— Modelling Frameworks – modelling Languages – Analysis modelling of web apps: Content; Interaction and Configuration models

UNIT III WEB APPLICATION DESIGN 9

Design for web apps – Goals – Design Process – Interactive Design: Principles and Guidelines; Workflow; Preliminaries; Design Steps; Usability and Issues – Design: Information; Navigation; Functional and Presentation

UNIT IV TESTING WEB APPLICATIONS 9

Testing fundamentals - Test specifics in Web Engineering - Test Approaches - Testing web Apps – Test schemes - Test methods and techniques — Test automation

UNIT V PROMOTING WEB APPLICATION AND WEB PROJECT MANAGEMENT 9

Operation and maintenance of web applications: Introduction; Challenges in launching the web application; Promoting web application; Content Management; Usage Analysis – Web project management –Usability of web application – Performance of web application.

TOTAL PERIODS: 45

OUTCOMES

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

- Understand web engineering concepts (K2)
- Compare different web application architectures and models (K4)
- Apply design technique to develop web applications (K3)
- Compare various testing approaches (K4)
- Apply guidelines to manage web applications (K3)

REFERENCE BOOKS

1. Gerti Kappel, Birgit Proll, Web Engineering, John Wiley and Sons Ltd, 2006. (unit I,II,III,IV,V)
2. Roger S. Pressman, David Lowe, Web Engineering, Tata McGraw Hill Publication, 2007. (unit I,II,III,IV)
3. Gustavo Rossi, Oscar Pastor, Daniel Schwabe, Luis Olsina, Web Engineering: Modelling and Implementing Web Applications, Springer, 2007. (unit II, III)
4. San Murugesan, Yogesh Deshpande, Web Engineering: Managing Diversity and Complexity of Web Application Development, Springer, 2001. (unit IV, V)

Course Code	Course Title	Category	L	T	P	C
PCP1278	NATURAL LANGUAGE PROCESSING	PE	3	0	0	3

OBJECTIVES

- To learn the language models.
- To understand the levels of knowledge in language processing.
- To explore text processing using Python.
- To understand the NLP applications

UNIT I OVERVIEW AND LANGUAGE MODELLING 9

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 WORLD LEVEL AND SYNTACTIC ANALYSIS 9

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 AND DISCLOSURE PROCESSING 9

The representation of Meaning: Meaning representation –Computational desiderata for representation; Lexical Semantics: word senses – relations – WordNet; Word Sense Disambiguation: Dictionary-based – Supervised – Minimally-supervised – Word Similarity: thesaurus methods – distributional methods; Discourse Processing: Reference resolution – Anaphora resolution algorithms – Co-reference resolution.

UNIT IV NATURAL LANGUAGE PROCESSING WITH PYTHON 9

Natural Language Toolkit (NLTK); Computing with Language: Texts and words – Simple statistics; Accessing Text Corpora: Brown corpus – Reuters corpus – Wordnet; Normalizing text – Automatic Tagging; Supervised Classification: Gender identification – Feature selection – Document classification.

UNIT V MACHINE TRANSLATION, IR AND IE 9

Machine Translation: Problems in machine translation – Classical MT – Statistical MT; Information Retrieval: The vector space model – Term weighting – Evaluation of IR; Information Extraction: Named entity recognition – Relation detection and classification.

TOTAL PERIODS: 45

OUTCOMES

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

- Describe the language models (K2)
- Explain levels of knowledge in language processing (K2)
- Write Python programs for text processing (K3)
- Apply NLP techniques to MT, IR and IE systems (K3)

REFERENCE BOOKS

1. Daniel Jurafsky, James H Martin, Speech and Language Processing: An introduction to Natural Language Processing, Computational Linguistics and Speech Recognition, Prentice Hall, 2nd Edition, 2008.
2. Tanveer Siddiqui, Tiwary U. S., Natural Language Processing and Information Retrieval, Oxford University Press, 2008.
3. Steven Bird, Ewan Klein, Edward Loper, Natural Language Processing with Python, O’Reilly, 2009.
4. Christopher D. Manning, Hinrich Schutze, Foundations of Statistical Natural Language Processing, MIT Press, 1999.
5. Nitin Indurkhya, Fred J. Damerau, Handbook of Natural Language Processing, CRC Press, 2nd Edition, 2010.

Course Code	Course Title	Category	L	T	P	C
PCP1222	WIRELESS NETWORKS	PE	3	0	0	3

OBJECTIVES

- To understand the fundamentals of wireless networks.
- To learn the concepts of various wireless technologies.
- To gain adequate exposure to the emerging technologies and their potential impact.

UNIT I WIRELESS LAN AND PAN 9

Introduction – fundamentals of WLAN: technical issues – network architecture; IEEE 802.11-physical layer – Mac layer mechanism– CSMA/CA,RTS/CTS – Polling – Bluetooth – Link manager protocol – L2CAP – Security – SDP – IEEE802.15.3.

UNIT II WIRELESS INTERNET 9

Introduction – wireless internet – address mobility – inefficiency of transport layer and Application layer protocol; mobile IP -– simultaneous binding – route optimization – mobile IP variations – handoffs – security in mobile IP; IP for wireless domain – TCP in wireless domain – TCP over wireless; impact of mobility.

UNIT III AD-HOC SENSOR NETWORK 9

Wireless Sensor Network – Applications – design challenges – Protocol stack – comparisons with MANET node architecture – network architecture – MAC protocols-requirements – IEEE 802.15.4 MAC protocol – Routing Protocol: energy aware routing – Location based routing – clustering – aggregation – QoS – security protocol – Zigbee standard.

UNIT IV 3G NETWORKS 9

Evolution from GSM – 3GServices and Applications –UMTS network structure – Core network – UMTS Radio access – HSPA – HSUPA – HSDPA – CDMA 1X – EVDO Rev -0, Rev-A, Rev-B, Rev-C Architecture – Protocol stack – Cognitive Radio network – Spectrum Sensing.

UNIT V 4G-LTE 9

Overview of LTE Networks – Need for LTE – From LTE to LTE-Advanced SAE: LTE Architecture – Radio Protocol stack – Interfaces; Concept of HetNET – Quality of Service and Bandwidth – Reservation – QoS metrics – Signalling for Bandwidth Requests and Grants – Bandwidth Allocation and Traffic Handling – Mobility Management – Security Protocols.

TOTAL PERIODS: 45

OUTCOMES

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

- Understand the design the wireless networks(K2)
- Analyze the wireless routing protocols(K3)
- Analyze the design of 3G networks(K3)
- Understand the design the 4G and LTE networks(K2)

REFERENCE BOOKS

1. Jochen Schiller, Mobile Communication, 2nd edition, Pearson education, 2005.
2. Holger Karl, Andreas Willing, Protocols and Architecture for Wireless Sensor Network, John Wiley & Sons, 2007.
3. Juha Korhonen, Introduction to 3G Mobile Communication, Artech House, 2003.
4. Harri Holma, Antti Toskala, HSDPA/HSUPA for UMTS, John Wiley & Sons, 2006.
5. Abd-Elhamid M. Taha, Hossam S. Hassanein, Najah Abu Ali, LTE, LTE-Advanced and Wimax towards IMT-advanced networks, John Wiley & Sons , 2012.
6. Vijay. K. Garg, Wireless Communication and Networking, Morgan Kaufmann Publishers, 2007.
7. Larry J. Greenstein, Andrea J. Goldsmith, Principles of Cognitive Radio, Cambridge University press, 2013.

Course Code	Course Title	Category	L	T	P	C
PCP1224	PRINCIPLES OF PROGRAMMING LANGUAGES	PE	3	0	0	3

OBJECTIVES

- To understand and describe syntax and semantics of programming languages.
- To understand Data, Data types, and Bindings.
- To understand programming language constructs.
- To explore object orientation, concurrency and event handling in programming languages.
- To explore the concepts of functional and logical programming.

UNIT I ELEMENTS OF PROGRAMMING LANGUAGES 9

Preliminaries: Concepts of programming languages – Language evaluation criteria – Influences on language design – Language categories – Implementation methods – Programming environments; Evolution of programming languages; Syntax and Semantics: Introduction – The general problem of describing syntax – Formal methods of describing syntax – Attribute grammars; Describing the meanings of programs: Dynamic semantics

UNIT II DATA TYPES AND ABSTRACTION 9

Bindings and Scopes: Names – Variables – The concept of binding – Scope – Scope and lifetime – Referencing environments –Named constants; Data Types: Primitive data types – Character string types – User - defined ordinal types – Array types – Associative arrays – Record types – Tuple types – List types – Union types – Pointer and reference types – Type checking – Strong typing – Type equivalence; Data Abstraction and Encapsulation: The Concept of abstraction – Parameterized abstract data types – Encapsulation constructs – Naming encapsulations.

UNIT III PROGRAMMING LANGUAGE CONSTRUCTS 9

Arithmetic expressions –Relational and Boolean expressions –Assignment statements – Mixed-mode assignments; Control Structures: Selection – Iterations – Branching – Guarded commands; Subprograms: Design issues – Local referencing – Parameter passing – Overloaded methods – Generic methods – Design issues for functions – User-defined overloaded operator – Closures and Coroutines

UNIT IV OBJECT ORIENTATION, CONCURRENCY & EVENT HANDLING 9

Object orientation: Design issues for OOP languages – Support for OOPs in C++, Java, C# and Ada; Concurrency: Subprogram-level concurrency – Semaphores – Monitors – Message passing – Threads – Statement level concurrency; Exception handling and Event handling.

UNIT V FUNCTIONAL & LOGIC PROGRAMMING 9

Functional Programming Languages: Mathematical functions – Fundamentals of functional programming languages – Scheme – Common LISP – Haskell; Concurrency in functional language; Logic Programming Languages: Introduction to predicate calculus – Proving theorems – Overview of logic programming – The origins of Prolog – Elements of Prolog – Deficiencies of Prolog – Applications of Logic Programming

TOTAL PERIODS: 45

OUTCOMES

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

- Describe syntax and semantics of programming languages. (K2)
- Explain data, data types, and constructs of programming languages. (K2)
- Explain object orientation, concurrency, exception and event handling in programming languages. (K2)
- Develop programs in LISP, ML, and Prolog. (K3)

REFERENCE BOOKS

1. Robert.W. Sebesta, Concepts of Programming Languages, Pearson Education, 10th Edition, 2012.
2. Carlo Ghezzi, Mehdi Jazayeri, Programming Language Concepts, John Wiley, 3rdEdition, 2008.
3. John C. Mitchell, Concepts in Programming Languages, Cambridge University Press, 2004.
4. Kenneth C. Louden, Kenneth A. Lambert, Programming Languages: Principles and Practice, Cengage Learning, 3rd Edition, 2012.

Course Code	Course Title	Category	L	T	P	C
PCP1225	INTRODUCTION TO CRYPTOGRAPHY	PE	3	0	0	3

OBJECTIVES

- To understand the classical and symmetric cryptographic techniques.
- To study about message authentication and hash functions.
- To acquire knowledge on 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 & SYMMETRIC CRYPTOGRAPHY 10

Cryptography and Modern Cryptography – Setting of Private-Key Encryption – Historical Ciphers and Their Cryptanalysis – The Basic Principles of Modern Cryptography; Perfectly Secret Encryption: Definitions and Basic Properties – The One-Time Pad – Limitations of Perfect Secrecy; Private-Key Encryption: A Computational Approach to Cryptography–Computationally Secure Encryption – Pseudo-randomness – Constructing Secure Encryption Schemes – Security Against Chosen-Plaintext Attacks (CPA) – Constructing CPA-Secure Encryption Schemes – Security Against Chosen-Ciphertext Attacks (CCA).

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 9

Substitution-Permutation Networks – Feistel Networks – DES – Increasing the Key Length of a Block Cipher – AES – Differential and Linear Cryptanalysis; One-Way Functions – From One-Way Functions to Pseudo randomness – Constructing Pseudorandom Generators – Constructing Pseudorandom Functions – Constructing Pseudorandom Permutations – Necessary Assumptions for Private-Key Cryptography.

UNIT IV NUMBER THEORY & KEY DISTRIBUTION 9

Number Theory: Preliminaries and Basic Group Theory – Primes, Factoring, and RSA – Assumptions in Cyclic Groups – 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.

TOTAL PERIODS: 45

OUTCOMES

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

- Describe and implement classical and symmetric ciphers (K2)
- Classify and discuss various attacks (K3)
- Compare and contrast symmetric and asymmetric encryption systems (K3)
- Describe the authentication schemes and hash algorithms (K2)
- Illustrate various Public key cryptographic techniques (K3)

REFERENCE BOOKS

1. Jonathan Katz, Yehuda Lindell, Introduction to modern cryptography, Chapman & Hall/CRC Cryptography and Network Security Series, Second Edition, 2014.
2. Wenbo Mao, Modern Cryptography – Theory and Practice, Pearson Education, 2004.
3. Johannes A. Buchmann, Introduction to Cryptography, Pearson Education, Springer, 2nd edition, 2009.
4. Charles P. Pfleeger, Shari Lawrence Pfleeger, Security in computing, Prentice Hall of India, Third Edition, 2006.
5. http://nptel.ac.in/courses/106105031/lecture by Dr. Debdeep Mukhopadhyay, IIT Kharagpur.

Course Code	Course Title	Category	L	T	P	C
PCP1226	**PARALLEL PROGRAMMING**	**PE**	**3**	**0**	**0**	**3**

OBJECTIVES

- To familiarize the issues in parallel computing.
- To describe distributed memory programming using MPI.
- To understand shared memory paradigm with Pthreads and with OpenMP.
- To learn the GPU based parallel programming using OpenCL.

UNIT I FOUNDATIONS OF PARALLEL PROGRAMMING 9

Motivation for parallel programming – Need Concurrency in computing; Basics of processes – multitasking – threads; — cache: cache mappings – caches and programs — virtual memory — Instruction level parallelism – hardware multi-threading; Parallel Hardware: SIMD — MIMD — Interconnection networks — cache coherence — Issues in shared memory model and distributed memory model; Parallel Software – Caveats – coordinating processes/ threads – hybrid model – shared memory model and distributed memory model; I/O; Performance of parallel programs; parallel program design.

UNIT II DISTIBUTED MEMORY PROGRAMMING WITH MPI 9

Basic MPI programming — $\mathrm{MPI}_{\mathrm{Init}}$ and $\mathrm{MPI}_{\mathrm{Finalize}}$ — MPI communicators – SPMD programs – $\mathrm{MPI}_{\mathrm{Send}}$ and $\mathrm{MPI}_{\mathrm{Recv}}$ — message matching; MPI: I/O – parallel I/O; Collective communication: Tree-structured communication – $\mathrm{MPI}_{\mathrm{Reduce}}$ — $\mathrm{MPI}_{\mathrm{Allreduce}}$ – broadcast – scatter – gather – allgather; MPI derived types -dynamic process management; Performance evaluation of MPI programs; A Parallel Sorting Algorithm

UNIT III SHARED MEMORY PARADIGM: PTHREADS 9

Basics of threads – Pthreads — Thread synchronization; Critical sections; Busy waiting; Mutex; Semaphores; Barriers and Condition Variables; Read-Write Lockswith examples; Caches, cache coherence and false sharing; Thread safety; Pthreads case study.

UNIT IV SHARED MEMORY PARADIGM: OPENMP 9

Basics OpenMP; Trapezoidal Rule; Scope of variables; Reduction clause; Parallel for directive; Loops in OpenMP; Scheduling loops; Producer Consumer problem; Cache issues; Threads safety in OpenMP; Two-body solvers; Tree Search

UNIT V GRAPHICAL PROCESSING PARADIGMS: OPENCL AND INTRODUCTION TO CUDA 9

Introduction to OpenCL — Example; OpenCL Platforms; Devices-Contexts; OpenCL programming: Built-In Functions – Programs Object and Kernel Object -– Memory Objects Buffers and Images – Event model -– Command-Queue – Event Object – Case study; Introduction to CUDA programming.

TOTAL PERIODS: 45

OUTCOMES

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

- Identify issues in parallel programming (K2)
- Develop distributed memory programs using MPI framework (K3)
- Design and develop shared memory parallel programs using Pthreads (K3)
- Design and develop shared memory parallel programs using OpenMP (K3)
- Implement Graphical Processing OpenCL programs (K3)

REFERENCE BOOKS

1. Peter S. Pacheco, An introduction to parallel programming, Morgan Kaufmann, 2011.
2. Munshi A., Gaster B., Mattson T. G., Fung J., Ginsburg D., OpenCL programming guide, Addison Wesley, 2011.
3. Quinn M. J., Parallel programming in C with MPI and OpenMP, Tata McGraw Hill, 2003.
4. Rob Farber, CUDA application design and development, Morgan Haufmann, 2011.
5. Gropp W., Lusk E., Skjellum A., Using MPI: Portable parallel programming with the message passing interface, MIT Press, Second Edition, 1999.

Course Code	Course Title	Category	L	T	P	C
PCP1227	SOFTWARE ARCHITECTURE AND DESIGN	PE	3	0	0	3

OBJECTIVES

- To understand the need, design approaches for software architecture to
- bridge the dynamic requirements and implementation.
- To learn the design principles and to apply for large scale systems.
- To design architectures for distributed heterogeneous systems, environment through brokerage interaction.
- To build design knowledge on service oriented and model driven architectures and the aspect-oriented architecture.
- To develop appropriate architectures for various Case studies like semantic web services, supply chain cloud services.

UNIT I SOFTWARE ARCHITECTURE BASICS 9

Introduction to Software Architecture – Bridging Requirements and Implementation – Design Guidelines: Software quality attributes – Software Architecture Design Space; Agile Approach to Software Architecture Design; Models for Software Architecture Description Languages (ADL).

UNIT II SOFTWARE ARCHITECTURE DESIGN PRINCIPLES 9

Object-Oriented Paradigm; Design Principles –Data centred software architecture – Repository architecture – Blackboard architecture; Hierarchical Architecture – Main-subroutine –Master-slave – Layered – Virtual machine; Interaction Oriented Software Architectures: Model-View-Controller (MVC), Presentation Abstraction-Control (PAC).

UNIT III DISTRIBUTED ARCHITECTURE AND ADAPTATION 9

Distributed Architecture: Client-Server, Middleware, Multi-tiers; Broker Architecture: MOM – CORBA – Message Broker Architecture - Service-Oriented Architecture (SOA): SOAP – UDDI – Grid/cloud service computing; Heterogeneous Architecture: Methodology of architecture decision, Quality attributes; Conceptual Framework For Architectural Adaptation: Techniques for supporting architecture centric change; The World Wide Web: A Case Study in Interoperability

UNIT IV SOFTWARE PRODUCT LINE ARCHITECTURE 9

Architecture of User Interfaces Containers – Case study – Web service; Product Line Architectures: Methodologies – Processes and tools; Software Reuse and Product Lines: Product line analysis – Design and implementation – Configuration models; Model Driven Architectures (MDA): Why MDA – Model transformation and software architecture – SOA and MDA – Eclipse modelling framework.

UNIT V ASPECT ORIENTED ARCHITECTURE 9

Aspect Oriented Architectures: AOP in UML – AOP tools – Architectural aspects and middleware selection of architectures; Evaluation of Architecture Designs; Case Study: Online computer vendor – Order processing – Manufacture & shipping - Inventory – Supply chain cloud service management; Semantic Web Services.

TOTAL PERIODS: 45

OUTCOMES

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

- Understand the need of software architecture for sustainable dynamic systems and adapt to the system according to the need (K2)
- Have a sound knowledge on design principles and to apply for large scale systems(K3)
- Design architectures for distributed heterogeneous systems(K3)
- Have good knowledge on service oriented and model driven architectures and the aspect-oriented architecture. (K2)
- Have a working knowledge to develop appropriate architectures through various case studies. (K2)

REFERENCE BOOKS

1. Ion Gorton, Essentials of Software Architecture, Springer Verlag, Second Edition, 2011.
2. Software Architecture Design Illuminated, Kai Qian Jones and Bartlett Publishers Canada, 2010.
3. David Budgen, Software Design, Pearson Education, Second Edition, 2004.
4. Len Bass, Paul Clements, Rick Kazman, Software Architecture in Practice, Addison, Wesley, Third Edition, 2012.
5. Mary Shaw, David Garlan, Software Architecture Perspectives on an emerging discipline, Pearson education, 2008.
6. Richard N. Taylor, Nenad Medvidovic, Eric M. Dashofy, Software Architecture, Foundations, Theory and Practice, Wiley 2010.

Course Code	Course Title	Category	L	T	P	C
PCP1279	IMAGE PROCESSING AND ANALYSIS	PE	3	0	0	3

OBJECTIVES

- To understand the basics of digital images.
- To understand the spatial and frequency domain processing.
- To learn basic image analysis - segmentation and feature detection.
- To understand colour image processing and image compression techniques.
- To appreciate the use of image processing in various applications.

UNIT I FUNDAMENTALS OF IMAGE PROCESSING 8

Introduction – Elements of visual perception – Steps in Image Processing Systems – Image Acquisition – Sampling and Quantization – Pixel Relationships – Image Modalities – File Formats – Image Operations: Arithmetic; Logical; Statistical and Spatial operations

UNIT II IMAGE ENHANCEMENT AND RESTORATION 10

Spatial Domain processing: Filtering operations; Histograms; Smoothing filters; Sharpening filters; Fuzzy techniques; Noise models; Filters for noise removal Frequency Domain processing: Fourier Transform – DFT and FFT; Filtering operations; Smoothing and Sharpening – Selective filters; Filters for noise removal; Homomorphic filtering Restoration: Model of Image Degradation/Restoration Process, Noise Models

UNIT III IMAGE SEGMENTATION AND FEATURE ANALYSIS 10

Thresholding techniques: Region growing; splitting and merging; Adaptive – Otsu method Edge detection: Template matching; Gradient operation; Hysterisis Thresholding – Canny operator – Laplacian operator; Image morphology – Binary and Gray Level morphology operations – erosion; dilation – opening– closing operations – Morphological watersheds; Features – Corner and interest point detection – boundary representation and detections – texture descriptors – regional descriptors and feature selection techniques

UNIT IV MULTI RESOLUTIONANALYSIS, COLOR IMAGES AND IMAGE COMPRESSIONS 9

Multi Resolution Analysis: Image Pyramids – Multi resolution expansion – Wavelet Transforms; Fast Wavelet transforms; Wavelet Packets Image Compression: Fundamentals – Models – Error Free Compression –Lossy Compression – Compression Standards – Watermarking Color Images: Color Models; Smoothing and Sharpening – Segmentation based on Color – Noise in Color Images

UNIT V CASE STUDIES IN IMAGE PROCESSING 8

Image Recognition : Fingerprint Recognition – Image Classification : Tumor classification from Medical Image – Image Understanding: CBIR – Image Fusion: Statellite image enhancement – Object tracking: Surveillance applications – Image Steganography: Image hiding in Multimedia.

TOTAL PERIODS: 45

OUTCOMES

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

- Design and implement enhancement and segmentation algorithms for image processing application. (K4)
- Perform analysis using various image features. (K3)
- Analyze the multi resolution techniques and methods used for color images. (K3)
- Make a positive professional contribution in the field of Digital Image Processing. (K4)

REFERENCE BOOKS

1. Rafael C. Gonzalez, Richard E. Woods, Digital Image Processing, Pearson Education, Third Edition, 2008. (Units I, II, III, IV)
2. Anil K.Jain, Fundamentals of Digital Image Processing, PHI, 2006.
3. Rafael C.Gonzalez, Richard E.Woods, Eddins, Digital Image Processing Using MATLAB, Tata McGraw-Hill, Second Edition, 2009.
4. Davis, E. R. Machine Vision Second Edition, 1997.

Course Code	Course Title	Category	L	T	P	C
PCP1228	SPEECH PROCESSING AND SYNTHESIS	PE	3	0	0	3

OBJECTIVES

- To explore the fundamentals of digital speech processing.
- To understand the basic concepts and algorithms of speech processing.
- To familiarize the students with the various speech signal representation, coding and recognition techniques.
- To study the concepts and evaluation method of speech synthesis

UNIT I FUNDAMENTALS OF DIGITAL SPEECH PROCESSING 9

Introduction: Discrete-Time signals and systems – Transform representation of Signals and systems – Fundamentals of digital filters – Sampling; Process of Speech Production – Acoustic theory of speech production – Digital models for speech signals.

UNIT II SPEECH SIGNAL ANALYSIS IN TIME DOMAIN 9

Time-dependent processing of speech – Methods for extracting the parameters: Energy – Average Magnitude – Zero-crossing rate; Silence discrimination using ZCR and energy – Short-time autocorrelation function – Pitch period estimation using autocorrelation function.

UNIT III SPEECH SIGNAL ANALYSIS IN FREQUENCY DOMAIN 9

Short time Fourier analysis – Fourier transform and linear interpretations – Sampling rates – Spectrographic Displays – Formant extraction – Pitch extraction – Linear predictive coding: Autocorrelation method – Covariance method; Solution of LPC equations – Durbin's Recursive solution – Application of LPC parameters – Pitch detection.

UNIT IV SPEECH RECOGNITION 9

Introduction – Pre-processing – Parametric representation – Speech segmentation – Dynamic time warping – Vector quantization – Hidden Markov Model – Language Models – Developing an isolated digit recognition system.

UNIT V SPEECH SYNTHESIS 9

Attributes of speech synthesis – Formant speech synthesis –Concatenative speech synthesis – Prosodic modification of speech – Source filter models for prosody modification – Evaluation of TTS system.

TOTAL PERIODS: 45

OUTCOMES

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

- Illustrate how the speech production is modelled (K2)
- Extract features from the speech signal in both time and frequency domain (K3)
- Developing a speech recognition system using statistical approach (K3)
- Compare the various methods of speech synthesis (K2)

REFERENCE BOOKS

- Rabiner L. R., Schaffer R. W., Digital Processing of Speech signals, Prentice Hall, 1978.
- Xuedong Huang, Alex Acero, Hsiao-Wuen Hon, Spoken Language Processing – A guide to Theory, Algorithm and System Development, Prentice Hall PTR, 2001.
- Lawrence Rabiner, Biing-Hwang Juang, Fundamentals of Speech Recognition, Prentice Hall Signal Processing Series, 1993.
- Thomas F.Quatieri, Discrete-Time Speech Signal Processing, Pearson Education, 2002.
- Ben Gold, Nelson Morgan, Speech and Audio Signal Processing, John Wiley and Sons Inc., Singapore, 2004.

Course Code	Course Title	Category	L	T	P	C
PCP1229	BIO INSPIRED COMPUTING	PE	3	0	0	3

OBJECTIVES

- To Learn bio-inspired theorem and algorithms.
- To understand foundations of complex systems and theoretical biology.
- To Understand random walk and simulated annealing.
- To Learn genetic algorithm and differential evolution.
- To Learn swarm optimization and ant colony for feature selection.
- To understand application of various algorithms.

UNIT I INTRODUCTION 9

Introduction to Algorithms: Newton’s method –Optimization –No-Free-Lunch theorems – Nature-Inspired metaheuristics; Analysis of Algorithms: Nature inspires algorithms – Parameter tuning and parameter control.

UNIT II RANDOM WALKS AND ANNEALING 9

Random Walks: Random variables – Isotropic random walks – Levy distribution and flights – Markov chains – Step sizes and search efficiency – Modality and intermittent search strategy – Importance of randomization – Eagle strategy; Annealing: Annealing and Boltzmann distribution – Parameters – SA algorithm – Stochastic tunnelling.

UNIT III GENETIC ALGORITHMS AND DIFFERENTIAL EVOLUTION 9

Introduction to genetic algorithms – Role of genetic operators – Choice of parameters – GA variants – Schema theorem – Convergence analysis; Introduction to differential evolution – Variants – Choice of parameters – Convergence analysis – Implementation.

UNIT IV SWARM OPTIMIZATION AND FIREFLY ALGORITHM 9

Swarm Intelligence – PSO algorithm – Accelerated PSO – Implementation – Convergence analysis – Binary PSO; The Firefly algorithms – Algorithm analysis – Implementation – Variants; Cuckoo behaviour – Cuckoo search.

UNIT V BAT FLOWER POLLINATION ANT AND BEE ALGORITHMS 9

Echolocation of bats – Bat algorithms – Binary Bat algorithms – Variants of the Bat algorithm; Flower pollination algorithms – Multi-Objective Flower pollination algorithms – validation and Numerical experiments; Ant algorithms – Bee-inspired algorithms.

TOTAL PERIODS: 45

OUTCOMES

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

- Understand principles of biologically inspired computing (K2)
- Understand random walk and simulated annealing (K2)
- Apply genetic algorithms (K3)
- Solve problems using swarm intelligence and cuckoo search (K3)
- Apply bio-inspired algorithms to solve real world problems (K3)

REFERENCE BOOKS

1. Xin-She Yang, Nature Ispired Optimization Algorithm, Elsevier, First Edition, 2014.
2. Eiben A.E., Smith, James E, Introduction to Evolutionary Computing, Springer, 2015.
3. Stephan Olariu, Albert Y. Zomaya, Handbook of Bioinspired Algorithms and Application, Chapman & Hall/CRC, 2006.
4. Dan Simon, Evolutionary Optimization Algorithms, John Wiley & Sons, 2013.
5. Helio J.C. Barbosa, Ant Colony Optimization - Techniques and Applications, Intech, 2013.
6. Yang, Cui, XIao, Gandomi, Karamanoglu ,Swarm Intelligence and Bio-Inspired Computing, Elsevier First Edition, 2013.

Course Code	Course Title	Category	L	T	P	C
PCP1231	BIO INFORMATICS	PE	3	0	0	3

OBJECTIVES

- To get exposed to the fundamentals of bioinformatics.
- To learn bio-informatics algorithm and phylogenetic concept.
- To understand open problems and issues in replication and molecular clocks.
- To learn to assemble genomes and corresponding theorems.
- To study and exposed to the domain of human genomics.

UNIT I INTRODUCTION AND FUNDAMENTALS 9

Fundamentals of genes, genomics, molecular evolution – genomic technologies – beginning of bioinformatics - genetic data – sequence data formats – secondary database – examples – data retrieval systems – genome browsers.

UNIT II BIOINFORMATICS ALGORITHM AND ANALYSIS 9

Sequence alignment and similarity searching in genomic databases: BLAST and FASTA – additional bioinformatics analysis involving nucleic acid sequences additional bioinformatics analysis involving protein sequences – Phylogenetic Analysis.

UNIT III DNA REPLICATION AND MOLECULAR CLOCKS 9

Beginning of DNA replication – open problems – multiple replications and finding replication – computing probabilities of patterns in a string-the frequency array-converting patterns – solving problems- finding frequents words-Big-O notation – case study: The Tower of Hanoi problem.

UNIT IV ASSEMBLE GENOMES AND SEQUNECES 9

Methods of assemble genomes – string reconstruction – De Bruijn graph – Euler’s theorem – assembling genomes –DNA sequencing technologies – sequence antibiotics – Brute Force Algorithm – Branch and Bound algorithm – open problems – comparing biological sequences-Case Study – Manhattan tourist Problem.

UNIT V HUMAN GENOME 9

Human and mouse genomes-random breakage model of chromosome evolution – sorting by reversals – greedy heuristic approach – break points- rearrangements in tumor and break point genomes-break point graphs – synteny block construction -open problems and technologies

TOTAL PERIODS: 45

OUTCOMES

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

- Understand the genomics technologies in Bioinformatics (K2)
- Deploy the replication and molecular clocks in bioinformatics. (K3)
- Understand how to assemble genomes and sequences. (K2)
- Apply the Microarray technologies for genome expression(K3).

REFERENCE BOOKS

1. Ion Mandoiu, Alexander Zelikovsky, Computational Methods for Next Generation Sequencing Data Analysis, Wiley series 2016.
2. Istvan Miklos, Renyi Institutue, Introduction to algorithms in bioinformatics, Springer 2016.
3. Philip Compeau and Pavel pevzner, Bioinformatics Algorithms: An Active Learning Approach, Cousera, Second edition volume I, 2015.
4. Supratim Choudhuri, Bioinformatics For Beginners, Elsevier, 2014.
5. Jin Xiong, Essential Bioinformatics, Cambridge University Press, 2007.

Course Code	Course Title	Category	L	T	P	C
PCP1232	**ADVANCED ALGORITHMS**	**PE**	**3**	**0**	**0**	**3**

OBJECTIVES

- To have a basic knowledge of computational tractability and intractability,
- To learn the concept of approximation algorithms, randomized algorithms and parallel algorithms,
- To understand the importance of local search in optimization problems.

UNIT I ALGORITHM ANALYSIS, NP AND COMPUTATIONAL INTRACTABILITY 9

Basics of Algorithm Analysis: Computational Tractability – Asymptotic Order of Growth; NP and Computational Intractability: Polynomial Time Reductions – The Satisfiability Problem – Efficient Certification and the Definition of NP – NP Complete Problems – Sequencing Problems – Partitioning Problems – Graph Coloring – Taxonomy of Hard Problems.

UNIT II APPROXIMATION ALGORITHMS 9

Approximation Algorithms: Greedy Algorithms and Bounds on the Optimum: A Load Balancing Problem – The Center Selection Problem – Set Cover: A General Greedy Heuristic – The Pricing Method: Vertex Cover – Arbitrarily Good Approximations: The Knapsack Problem.

UNIT III LOCAL SEARCH 9

Local Search: The Landscape of an Optimization Problem – The Metropolis Algorithm and Simulated Annealing – An Application of Local Search to Hopfield Neural Networks – Maximum-Cut Approximation via Local Search – Choosing a Neighbor Relation.

UNIT IV RANDOMIZED ALGORITHMS 9

Randomized Algorithms: Basic Probability Definitions – A First Application: Contention Resolution – Finding the Global Minimum Cut – Random Variables and Their Expectations – A Randomized Approximation Algorithm for MAX 3-SAT – Randomized Divide and Conquer: Median-Finding and Quicksort – Hashing: A Randomized Implementation of Dictionaries.

UNIT V PARALLEL ALGORITHMS 9

Introduction: Parallel Processing – Parallel Models – Performance of Parallel algorithms; Searching, Merging and Sorting: Searching; Graphs: Connected Components – Minimum Spanning Trees.

TOTAL PERIODS: 45

OUTCOMES

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

- Understand the importance of computational tractability and intractability in algorithm analysis (K3).
- Understand and differentiate between approximation algorithms, randomized algorithms and parallel algorithms (K3).
- Apply local search in optimization problems(K3).

REFERENCE BOOKS

1. Kleinberg J, Tardos E, Algorithm Design, Addison-Wesley, 2005.
2. Joseph Jaja, An Introduction to Parallel Algorithms, Addison Wesley, 1992.
3. Akl S., Design and Analysis of Parallel Algorithms, Prentice Hall Inc, 1992.

Course Code	Course Title	Category	L	T	P	C
*******	NETWORK SECURITY	PE	3	0	0	3

OBJECTIVES

- To understand the various attacks.
- To study about an essential of computer security.
- To acquire knowledge on TCP/IP security, firewalls, IPSec, Virtual Private Networks, intrusion detection systems.
- To understand how various security mechanisms work and correlate these security mechanisms with security principles.
- To apply security principles to solve problems.

UNIT I TCP ATTACKS 9

Packet Sniffing and Spoofing: Packet Sniffing – Packet Spoofing – Snoofing: Sniffing and Spoofing; Attacks on the TCP Protocol: TCP Protocol Working – SYN Flooding Attack – TCP Reset Attack – TCP Session Hijacking Attack.

UNIT II AUTHENTICATION AND STANDARDS 9

Authentication: Authentication Systems – Authentication of People – Symmetric Key Distribution Using Symmetric Encryption – Kerberos V4, V5.

UNIT III PUBLIC KEY INFRASTRUCTURE 9

Attack on Public Key Cryptography – Public Key Certificates: X.509 Authentication services – Attacks on PKI – Types of Digital Certificates.

UNIT IV SECURITY PRACTICE & SYSTEM SECURITY 9

Internet Firewalls for Trusted System: Roles of Firewalls – Types of Firewalls – Netfilter – Iptables – Firewall design Principles – DNS Attacks – Cache Poisoning – SET for E-Commerce Transactions. Intruder – Intrusion detection system – Virus and related threats – Countermeasures.

UNIT V 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-Nonrepudiation – Pretty Good Privacy – S/MIME; IP Security: Overview of IPSec – IP and IPv6 – Authentication Header – Encapsulation Security Payload (ESP) – Internet Key Exchange (Phases of IKE, ISAKMP/IKE Encoding);Web Security: SSL/TLS Basic Protocol – Computing the keys – Client authentication – PKI as deployed by SSL Attacks fixed in v3 – Exportability-Encoding-Secure Electronic Transaction (SET).

TOTAL PERIODS: 45

OUTCOMES

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

- Classify and discuss various attacks (K3).
- Design Secure applications (K3).
- Inject secure coding in the developed applications (K3).

REFERENCE BOOKS

1. William Stallings, Network Security Essentials: Applications and Standards, Pearson, Sixth Edition, 2017.
2. Charlie Kaufman, Radia Perlman, Mike Speciner, Network Security: Private Communication in a Public World, Pearson Education, Second Edition, 2017.
3. Wenliang Du, Computer Security: A Hands-on Approach, CreateSpace Independent Publishing Platform, First Edition, 2017.

Course Code	Course Title	Category	L	T	P	C
PCP1376	REAL TIME SYSTEMS	PE	3	0	0	3

OBJECTIVES

- To learn real time operating system concepts, the associated issues & Techniques.
- To understand design and synchronization problems in Real Time System.
- To learn how communication is done in real time systems.
- To explore the concepts of real time databases.
- To understand the evaluation techniques, present in the Real Time System.

UNIT I REAL TIME SYSTEMS AND SCHEDULING 9

Introduction – Structure of a real time system – Task classes – Performance measures for real time systems – Estimating program run times – Issues in real time computing – Task assignment and scheduling –Classical uniprocessor scheduling algorithms – Fault tolerant scheduling.

UNIT II INTERTASK COMMUNICATION AND MEMORY MANAGEMENT 9

Buffering data – Time relative buffering – Ring buffers; Mailboxes –Queues – Critical regions – Semaphores – Other Synchronization mechanisms – Deadlock – Priority inversion – Process stack management –Run time ring buffer – Maximum stack size – Multiple stack arrangement – Memory management in task control block – Swapping – Overlays – Block page management – Replacement algorithms – Memory locking – Working sets – Real time garbage collection – Contiguous file systems.

UNIT III REAL TIME COMMUNICATION 9

Basic concepts – Real time communication in a LAN – Soft and hard real time communication in a LAN – Bounded access protocol – Real time communication over internet – Routing – Multicast routing – Resource reservation – Traffic shaping and policing – Scheduling mechanisms.

UNIT IV REAL TIME DATABASES 9

Real Time Databases: Basic definition – Real time vs general purpose databases – Main memory databases; Transaction priorities – Transaction aborts – Concurrency control issues; Disk Scheduling Algorithms – Two– phase approach to improve predictability – Maintaining Serialization consistency – Databases for hard real time systems.

UNIT V EVALUATION TECHNIQUES AND CLOCK SYNCHRONIZATION 9

Reliability Evaluation Techniques: Obtaining parameter values – Reliability models for hardware redundancy – Software error models; Clock Synchronization: Clock, A Nonfault–tolerant synchronization algorithm – Impact of faults – Fault tolerant synchronization in hardware – Fault tolerant synchronization in software).

TOTAL PERIODS: 45

OUTCOMES

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

- Apply principles of real time system design techniques to develop real time applications. (K3)
- Demonstrate communication and memory management principles between real time tasks. (K3)
- Apply network communication principles in real time systems. (K3)
- Apply database concepts in real time applications. (K3)
- Employ evaluation techniques for any real time application. (K3)

REFERENCE BOOKS

1. Krishna C. M., Kang G. Shin, Real-Time Systems, McGraw-Hill International Editions, 2010. (Units 1,4,5)
2. Phillip. A. Laplante,Seppo J Ovaska, Real-Time System Design and Analysis: Tools for the Practitioner, Wiley, 4th Edition, 2013 (Unit 2)
3. Rajib Mall, Real-time systems: theory and practice, Pearson Education, 2009. (Unit 3)
4. Buhur R. J. A., Bailey D. L., An Introduction to Real-Time Systems, Prentice Hall International, 1999.
5. Stuart Bennett, Real Time Computer Control - An Introduction, Prentice Hall of India, 1998
6. Allen Burns, Andy Wellings, Real Time Systems and Programming Languages, Pearson Education, 2003.

Course Code	Course Title	Category	L	T	P	C
PCP1321	SOFTWARE PROJECT MANAGEMENT	PE	3	0	0	3

OBJECTIVES

- Understand the basic concept of project management.
- Learn the various costing and life cycle management.
- Understand the role played by risk in software project.
- Appreciate the use of metrics for software project management.
- Know the challenges in people management.

UNIT I PROJECT MANAGEMENT AND COSTING 9

Software Project Management: Approaches - Project acquisition - Initiation - Planning - Execution - Execution control - Change management - Scheduling - Project closure.

UNIT II LIFE CYCLE MODELS AND QUALITY MANAGEMENT 9

Introduction to life cycle models - Agile development models - Extreme programming - Scrum; Project planning - Metrics of project size estimation – Project estimation techniques - Empirical estimation techniques - COCOMO – Halstead - Staffing level; ISO 9000 - SEI CMM -SPICE - PSP.

UNIT III RISK MANAGEMENT 9

Perspectives of Risk Management - Risk Definition - Risk Categories – Risk Assessment - Approaches, techniques and good practices - Risk Identification - Analysis - Prioritization - Risk Control - Planning – Resolution - Monitoring - Risk Retention – Risk Transfer – Failure Mode and Effects Analysis (FMEA) - Operational Risks – Supply Chain Risk Management.

UNIT IV METRICS 9

Need for software metrics – Scope – Basics – Framework for software measurement - Classification of software metrics: Product metrics - Size metrics – Complexity metrics - Halstead’s product metrics - Quality metrics - Process metrics - Empirical models - Statistical models - Theory-based models - Composite models - Reliability models - Measuring internal and external product attributes.

UNIT V PEOPLE MANAGEMENT 9

Managing people in software environment: Introduction - Understanding behaviour-Organizational behaviour- Selecting the right person- Oldham-Hackman job characteristics model; Stress: Stress management - Health and safety – Ethical and professional concerns; Working in teams: Becoming a team – Decision making - Organization and team structure - Coordination dependencies – Dispersed and virtual teams - Communication genres - Communication plans -Leadership.

TOTAL PERIODS: 45

OUTCOMES

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

- Identify the various elements of software management process framework (K2)
- Use available open-source estimation tools for cost estimation (K3)
- Identify existing risk and perform risk assessment (K2)
- Design a software metric for software project management (K3)
- Modify the art of interviewing people for a given scenario (K3)

REFERENCE BOOKS

1. Murali Chemuturi, Thomas M. Cagley, Mastering Software Project Management: Best Practices, Tools and Techniques, J. Ross Publishing, 2010. (Unit 1)
2. Rajib Mall, Fundamentals of Software Engineering, PHI Learning Pvt. Ltd., 3rd edition, 2009. (Unit 2)
3. Ravindranath Pandian, Applied Software Risk Management A Guide for Software Project Managers, Auerbach Publication, 2007. (Unit 3)
4. Antonio Borghesi, Barbara Gaudenzi, Risk Management: How to Assess, Transfer and Communicate Critical Risks: Perspectives in Business Culture, Springer, Illustrated Edition, 2012. (Unit 3)
5. Norman Fenton, James Bieman, Software Metrics: A Rigorous and Practical Approach, CRC Press, 3rd edition, 2015.
6. Bob Hughes, Mike Cotterell, Rajib Mall, Software Project Management, Tata McGraw Hill, Sixth Edition, 2018.

Course Code	Course Title	Category	L	T	P	C
PCP1322	**COMPUTER VISION**	**PE**	**3**	**0**	**0**	**3**

OBJECTIVES

- To review image processing techniques for computer vision
- To understand shape and region analysis
- To understand Hough, Transform and its applications to detect lines, circles, ellipses.
- To understand image registration and fusion techniques
- To understand 3D image visualization
- To understand motion analysis
- To study some applications of computer vision algorithms

UNIT I IMAGE PROCESSING FUNDAMENTALS 9

Review of Image Processing Techniques: Classical filtering operations; Thresholding techniques – Segmentation: Edge detection techniques – Corner and interest point detection; Mathematical morphology; Texture.

UNIT II SHAPES AND REGIONS 9

Binary Shape Analysis: Connectedness – Object labelling and counting – Size filtering –Distance functions – Skeletons and thinning – Deformable shape analysis; Boundary Tracking Procedures: Active contours – Shape models and shape recognition – Centroidal profiles – Handling occlusion – Boundary length measures; Boundary Descriptors: Chain codes; Fourier Descriptors; Region Descriptors; Moments.

UNIT III HOUGH TRANSFORM, IMAGE REGISTRATION AND FUSION 9

Line Detection: Hough Transform (HT) for line detection – RANSAC for straight line detection; HT based circular object detection; HT based ellipse detection; Image Registration: Pre-processing – Feature selection – Feature correspondence – Transformation functions.

UNIT IV 3D VISION 9

Sources of 3D Data sets: Slicing the data set – Arbitrary section planes; The use of color; Volumetric Display: Stereo viewing – Ray tracing – Reflection – Surfaces – Multiply connected surfaces; Methods for 3D vision: Projection schemes – Shape from shading – Photometric stereo – Shape from texture – Shape from focus – Active range finding; Surface Representations; Point-based Representations; Volumetric Representations.

UNIT V 3D MOTION AND APPLICATIONS 9

Introduction to Motion: Triangulation – Bundle adjustment – Translational alignment – Parametric motion – Spline-based motion – Optical flow – Layered motion; Application: Photo album – Face detection – Face recognition – Eigen faces – Active appearance and 3D shape models of faces; Application: Surveillance – Foreground - background separation – Particle filters – Chamfer matching – Tracking and occlusion – Combining views from multiple cameras – Human gait analysis; Application: In -vehicle vision system – locating roadway – road markings – identifying road signs – locating pedestrians.

TOTAL PERIODS: 45

OUTCOMES

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

- Implement fundamental image processing techniques required for computer vision (K4)
- Perform shape analysis and Implement boundary tracking techniques (K3)
- Apply 3D vision techniques (K3)
- Develop applications using computer vision techniques (K4)

REFERENCE BOOKS

1. Rafael C.Gonzalez and Richard E.Woods, Digital Image Processing, Pearson Education, Third Edition, 2008.
2. Davies E. R., Computer & Machine Vision, Academic Press, Fourth Edition, 2012.
3. Szeliski R., Computer Vision: Algorithms and Applications, Springer 2011.
4. Mark Nixon, Alberto S. Aquado, Feature Extraction & Image Processing for Computer Vision, Academic Press, Third Edition, 2012.
5. John C.Russ, The Image Processing Handbook, CRC Press, 2007.
6. Baggio D. L., Mastering OpenCV with Practical Computer Vision Projects, Packt Publishing, 2012.
7. Jan Erik Solem, Programming Computer Vision with Python: Tools and Algorithms for Analyzing Images, O'Reilly Media, 2012.

Course Code	Course Title	Category	L	T	P	C
PCP1377	INFORMATION RETRIEVAL TECHNIQUES	PE	3	0	0	3

OBJECTIVES

- To understand the basics of information retrieval with pertinence to modelling – query operations and indexing
- To get an understanding of machine learning techniques for text classification and clustering
- To understand the various applications of information retrieval giving emphasis to multimedia IR – web search and digital libraries
- To explore various IR applications using case studies

UNIT I INTRODUCTION AND MODELLING 9

Basic Concepts: Retrieval process – Architecture – Boolean retrieval; IR Models: Taxonomy and characterization of IR models – Classical IR models – Alternative set theoretic models – Alternative algebraic models; Models for Browsing; Retrieval Evaluation: Performance evaluation – Reference collection.

UNIT II INDEXING AND QUERYING 9

Indexing: Inverted indices – Suffix trees – Suffix arrays – Compression; Querying: Query languages – Query operations: Relevance feedback and query expansion - Automatic local and global analysis; Text Properties; Text Operations: Text preprocessing – Text compression.

UNIT III SEARCHING 10

Searching: Sequential searching – Pattern matching; Searching the Web: Characterizing The Web – Search engines – Browsing – Searching using hyperlinks.

UNIT IV CLASSIFICATION AND CLUSTERING 9

Text Classification: Naive Bayes; Vector Space Classification: Rocchio – k- Nearest Neighbour; Flat Clustering: K-Means – Model-based clustering; Hierarchical Clustering; Matrix Decompositions and Latent Semantic Indexing.

UNIT V APPLICATIONS 8

XML Retrieval – Multimedia IR – Parallel and Distributed IR – Digital Libraries – Social Media Retrieval – Content-based Image Retrieval.

TOTAL PERIODS: 45

OUTCOMES

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

- Explain various IR modelling techniques. (K2)
- Identify and design the various components of an Information Retrieval system. (K3)
- Apply machine learning techniques to text classification and clustering which is used for efficient Information Retrieval. (K3)
- Explain various IR applications. (K2)

REFERENCE BOOKS

1. Ricardo Baeza-Yates, Berthier Ribeiro Neto, Modern Information Retrieval: The concepts and Technology behind Search, ACM Press Books, Second Edition, 2011.
2. Christopher D. Manning, Prabhakar Raghavan, Hinrich Schutze, Introduction to Information Retrieval, Cambridge University Press, First South Asian Edition, 2008.
3. Stefan Buttcher, Charles L. A. Clarke, Gordon V. Cormack, Information Retrieval - Implementing and Evaluating Search Engines, The MIT Press, Cambridge, Massachusetts London, England, 2010.

Course Code	Course Title	Category	L	T	P	C
PCP1378	DATA VISUALIZATION TECHNIQUES	PE	3	0	0	3

OBJECTIVES

- To develop skills to both design and critique visualizations.
- To introduce visual perception and core skills for visual analysis.
- To understand visualization for time-series, ranking analysis and deviation analysis.
- To understand visualization for distribution analysis, correlation analysis and multivariate analysis.
- To understand issues and best practices in information dashboard design.

UNIT I CORE SKILLS FOR VISUAL ANALYSIS 9

Information Visualization – Effective Data Analysis – Traits of Meaningful Data – Visual Perception – Making Abstract Data Visible – Building Blocks of Information Visualization – Analytical Interaction – Analytical Navigation – Optimal Quantitative Scales – Reference Lines and Regions – Trellises and Crosstabs – Multiple Concurrent Views – Focus and Context – Details on Demand – Overplotting reduction – Analytical Patterns: Pattern examples.

UNIT II TIME-SERIES, RANKING, AND DEVIATION ANALYSIS 9

Time-series Analysis: Time-series patterns – Time-series displays – Time-series analysis techniques and best practices; Part-to-whole and Ranking Analysis: Part-to-whole and ranking patterns – Part-to-whole and ranking displays – Part-to-whole and ranking techniques and best practices; Deviation Analysis: Deviation analysis displays – Deviation analysis techniques and best practices.

UNIT III DISTRIBUTION, CORRELATION, AND MULTIVARIATE ANALYSIS 9

Distribution Analysis: Describing distributions – Distribution patterns – Distribution displays – Distribution analysis techniques and best practices; Correlation Analysis: Describing correlations – Correlation patterns – Correlation displays – Correlation analysis techniques and best practices; Multivariate Analysis: Multivariate patterns – Multivariate displays – Multivariate analysis techniques and best practices.

UNIT IV INFORMATION DASHBOARD DESIGN 9

Information Dashboard: Clarifying the vision – Thirteen common mistakes in dashboard design – Tapping into the power of visual perception – Eloquence through simplicity – Effective dashboard display media – Putting it all together; Dashboard design using Shiny.

UNIT V DATA VISUALIZATION TOOLS 9

Tableau: Introducing the tableau desktop workspace – Connecting to your data – Building your first visualization – Creating a standard map view – Plotting your own locations on a map – Building your first advanced dashboard.

TOTAL PERIODS: 45

OUTCOMES

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

- Recognize core skills for visual analysis (K2)
- Apply visualization techniques for various data analysis tasks (K3)
- Employ information dashboard design techniques (K3)
- Use different tools to better visualize data (K3)

REFERENCE BOOKS

1. Stephen Few, Now you see it: Simple Visualization techniques for quantitative analysis, Analytics Press, 2009. (units I, II, III)
2. Stephen Few, Information dashboard design: Displaying data for at-a-glance monitoring, Analytics Press, second edition, 2013. (unit IV)
3. Daniel G Murray, Tableau your data!: Fast and easy visual analysis with Tableau software, Wiley, second edition, 2016. (unit V)
4. Ben Fry, Visualizing data: Exploring and explaining data with the processing environment, O'Reilly, 2008.
5. Edward R. Tufte, The visual display of quantitative information, Graphics Press, Second Edition, 2001.
6. Nathan Yau, Data Points: Visualization that means something, Wiley, 2013.
7. Tamara Munzner, Visualization Analysis and Design, AK Peters Visualization Series, CRC Press, Nov. 2014.

Course Code	Course Title	Category	L	T	P	C
PCP1323	**SOFTWARE DEFINED NETWORKS**	**PE**	**3**	**0**	**0**	**3**

OBJECTIVES

- To learn about the concepts, basics of SDN and Openflow.
- To understand SDN Controller and programming with SDN.
- To learn about various data centers and about Network Function Virtualization.
- To study Network Topology & SDN Framework.
- To investigate various use cases using SDN.

UNIT I SDN PLANE SEPARATION & OPENFLOW 9

Separation of Control Plane and Data Plane – Distributed and centralized control plane – OpenFlow protocol – Hybrid Approaches.

UNIT II SDN CONTROLLER AND PROGRAMMING SDN 9

SDN Controller: General Concepts – Layer3 Centric – Plexi – CiscoOnePK ; Programming in SDN : The Management Interface – Application-Network Divide – Modern Orchestration.

UNIT III DATA CENTER & NETWORK FUNCTION VIRTUALIZATION 9

Data Center: Multitenant Data Center – Virtualised Multitenant Data Center – SDN solution for Data Center – VLAN – NVGRE; Network Function Virtualization: Virtualization and Data Plane I/O - Service Locations and Chaining - NFV at ETSI and Non-ETSI.

UNIT IV NETWORK TOPOLOGY & SDN FRAMEWORK 9

Network Topology: Traditional methods, LLDP, BGP - TE/LS, ALTO, I2RS Topology; SDN Framework: Juniper SDN framework – IETF SDN Frameworks – Open Daylight Controller Framework.

UNIT V USE CASES 9

Use cases for bandwidth scheduling – manipulation and calendaring – Use Cases for Data Center Overlays – Big Data – and Network Function Virtualization – Input traffic classification and triggered actions.

TOTAL PERIODS: 45

OUTCOMES

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

- Understand the concepts, basics of SDN and Openflow (K2)
- Understand SDN Controller and programming with SDN (K2)
- Analyze various SDN solutions for data center and about Network Function Virtualization. (K3)
- Design an SDN Framework (K2)
- Analyze about various use cases in implementing SDN(K3)

REFERENCE BOOKS

1. Thomas D. Nadeau, SDN: Software Defined Networks, An Authoritative Review of Network Programmability Technologies, Ken Gray Publisher: O'Reilly Media, Third Edition, 2013.
2. Paul Goransson and Chuck Black, Software Defined Networks: A Comprehensive Approach, Morgan Kaufmann, 2014.
3. Vivek Tiwari, SDN and OpenFlow for Beginners Amazon Digital Services, Inc., 2013.
4. Fei Hu, Network Innovation through OpenFlow and SDN: Principles and Design, CRC Press, 2014.

Course Code	Course Title	Category	L	T	P	C
PCP1324	LOGIC IN COMPUTER SCIENCE	PE	3	0	0	3

OBJECTIVES

- To understand the need of logic for specification and verification of computer systems.
- To learn the syntax and semantics of various logics.
- To learn the skill of writing formal specifications in various logics.
- To learn resolution for propositional and predicate logic.
- To learn model checking algorithm for LTL and CTL.
- To learn PROLOG.

UNIT I PROPOSITIONAL LOGIC 9

Foundations: Syntax and Semantics – Mathematical Induction – Soundness and Completeness; Semantic Entailment: Natural Deduction; Normal Forms; SAT; Resolution; Horn Logic; DPLL; CDCL; SAT Solvers; Binary Decision Diagrams.

UNIT II PREDICATE LOGIC 9

Foundations: Syntax and Semantics; Semantic Entailment: Natural Deduction; Normal Forms; Undecidability of Predicate Logic; Herbrand’s Theory; Resolution: Ground Resolution – Unification; Refinements of Resolution: P-resolution – N-resolution – linear resolution – unit resolution – SLD resolution – LUSH resolution.

UNIT III TEMPORAL LOGICS 9

Foundations; Linear-time temporal logic; Syntax of LTL–Semantics of LTL – Practical patterns of specifications; Branching-time logic: Syntax of CTL – Semantics of CTL – Practical patterns of specifications; CTL* and the expressive powers of LTL and CTL; Model-checking algorithms: The CTL model-checking algorithm – CTL model checking with fairness – The LTL model-checking algorithm; NuSMV.

UNIT IV 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; PROLOG.

UNIT V PROGRAM VERIFICATION 9

Foundations; A framework for software verification: A core programming language – Hoare triples – Partial and total correctness – Program variables and logical variables; Proof calculus for partial correctness: Proof rules – Proof tableaux – A case study: minimal-sum section; Proof calculus for total correctness; Programming by contract.

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 logic programs in PROLOG. (K3)
- Prove Hoare triples for simple programs. (K3)

REFERENCE BOOKS

1. Huth M. and Ryan M., Logic in Computer Science–Modeling and Reasoning About Systems, Cambridge University Press, Second Edition, 2004. (Units I,II,III and IV)
2. Uwe Schoning Logic for Computer Scientists, Birkhauser, 1989. (Units I,II,IV)
3. Ben-Ari M., Mathematical Logic for Computer Science, Springer, Second Edition, 2003.

Course Code	Course Title	Category	L	T	P	C
PCP1325	MOBILE AND PERVASIVE COMPUTING	PE	3	0	0	3

OBJECTIVES

- To learn the basic mobile concepts.
- To understand the 4G Telecommunication System Principles.
- To introduce the broad perspective of pervasive concepts and management.
- To explore the HCI in Pervasive environment.
- To apply the pervasive concepts in mobile environment.

UNIT I INTRODUCTION 9

History; Telecommunications: GSM – GPRS – DECT – TETRA – UMTS and IMT- 2000; Wireless LAN: WiFi, Bluetooth,WiMAX,Wireless ATM.

UNIT II OVERVIEW OF A MODERN 4G TELECOMMUNICATIONS SYSTEM 9

Introduction – LTE-A System Architecture: LTE RAN – OFDM Air Interface – Evolved Packet Core – LTE Requirements – LTE-Advanced – Enhancements in LTE-A; Introduction to 5G.

UNIT III PERVASIVE COMPUTING CONCEPTS AND ELEMENTS 9

Perspectives of Pervasive Computing – Challenges – Technology – Infrastructure and Devices – Middleware for Pervasive Computing Systems – Pervasive Computing Environments – Context Collection and Wireless Sensor Networks – User Tracking – Context Reasoning.

UNIT IV HUMAN COMPUTER INTERFACE IN PERVASIVE ENVIRONMENTS 9

HCI Service and Interaction Migration – Context Driven HCI Service Selection – Interaction Service Selection Overview – User Devices – Service-Oriented Middleware Support – User History and Preference – Context Manager – Local Service Matching – Global Combination Selection – Effective Region – User Active Scope – Service Combination Selection Algorithm – A web based HCI migration framework.

UNIT V PERVASIVE MOBILE TRANSACTION 9

Introduction to Pervasive Transactions – Mobile Transaction Framework – Context- Aware Pervasive Transaction Model – Dynamic Transaction Management – Formal Transaction Verification.

TOTAL PERIODS: 45

OUTCOMES

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

- Understand the Basic architecture and concepts of Mobile Computing (K2)
- Understand the 4G Telecommunication System Principles (K2)
- Understand the basis of pervasive computing. (K2)
- Apply HCI in Pervasive environment. (K3)
- Model pervasive computing problems in mobile environment. (K3)

REFERENCE BOOKS

1. Schiller J., Mobile Communications, Addison Wesley, 2000. (Unit I)
2. Minyi Guo, Jingyu Zhou, Feilong Tang, Yao Shen, Pervasive Computing: Concepts, Technologies and Applications CRC Press, 2016 (Unit III, IV, V)
3. Juha Korhonen, Introduction to 4G Mobile Communications, Artech House Publishers, 2014. (Unit II)
4. Kolomvatsos, Kostas, Intelligent Technologies and Techniques for Pervasive Computing, IGI Global, 2013.
5. Bala Krishna M., Jaime Lloret Mauri, Advances in Mobile Computing and Communications: Perspectives and Emerging Trends in 5G Networks, CRC 2016
6. Alan Colman, Jun Han, Muhammad Ashad Kabir, Pervasive Social Computing Socially-Aware Pervasive Systems and Mobile Applications, Springer, 2016.

Course Code	Course Title	Category	L	T	P	C
PIF1176	AGILE SOFTWARE ENGINEERING	PE	3	0	0	3

OBJECTIVES

- Understand agile software development practices.
- Demonstrate Agile development and testing techniques.
- Know the benefits and pitfalls of working in an Agile team.
- Understand agile development and testing.

UNIT I AGILE SOFTWARE DEVELOPMENT 9

Agile software development; Traditional software development – waterfall approach – requirements – design phase – implementation – testing – support – advantages and disadvantages; incremental development; Lean software development; scrum; test driven development; extreme programming; rational unified process; agile unified process; Agile model driven development.

UNIT II AGILE XP PRODUCT DEVELOPMENT 9

People and Teams working together – Team behaviour in XP projects – setting up a team – developing team skills – training together; organizational framework; planning – PERT – Gantt charts; Dealing with problem – basic strategies – when things go really wrong; risk analysis; starting XP project – business analysis and problem discovery; techniques for requirement elicitation; Developing the requirement documents.

UNIT III BUILDING USER STORIES AND TESTING 9

Identifying stories and preparing to build – looking at the user stories; collection of stories; user interfaces; communicating clearly with the customer and building confidence; demonstrating the non-functional requirements – non-functional requirements; Estimating resources – software cost estimation – object point analysis – COSMIC FFP – Design for test; Automating unit tests – writing unit tests in JUnit – managing tests.

UNIT IV AGILE PRACTICE VALIDATION 9

Study Design – Survey field study – Questionnaire Design – Data collection procedure; Construct operationalization – Adoption of agile practices – teamwork and contextual variables – team performance – instrument validation; Analysis methods – regression analysis – structural equation modelling – selecting appropriate analysis techniques; Assessment of team performance; Hypotheses: Test and Evaluation – measurement model effects on team potency and team performance.

UNIT V AGILITY AND QUALITY ASSURANCE 9

Agile Interaction Design – Agile product development – Agile Metrics – Feature Driven Development (FDD) – Financial and Production Metrics in FDD – Agile approach to Quality Assurance – Test Driven Development – Pair programming: Issues and Challenges – Agile approach to Global Software Development.

TOTAL PERIODS: 45

OUTCOMES

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

- Define the interaction with business stakeholders in determining the requirements for a software system. (K2)
- Apply iterative software development process. (K3)
- Determine the impact of social aspects on software development success. (K3)
- Explain the approach of product development using Agile. (K2)
- Prepare the solution using agile knowledge sharing methods. (K3)

REFERENCE BOOKS

1. Thomas Stober, Uwe Hansmann, Agile Software Development: Best Practices for Large Software Development Projects, Springer-Verlag Berlin Heidelberg 2010. (Unit 1)
2. Mike Holcombe, Running an Agile Software Development Project, University of Sheffield, United Kingdom, JohnWiley &amp; Sons, Inc., Hoboken, New Jersey, 2008. (Unit 2,3)
3. Christoph Schmidt, Agile Software Development Teams, Springer International Publishing Switzerland 2016. (Unit 4)
4. Dingsoyr, Torgeir, Dyba, Tore, Moe, Nils Brede (Eds.), Agile Software Development, Current Research and Future Directions?, Springer-Verlag Berlin Heidelberg, 2010 (Unit 5)

Course Code	Course Title	Category	L	T	P	C
PCP1326	**SOCIAL NETWORK ANALYSIS**	**PE**	**3**	**0**	**0**	**3**

OBJECTIVES

- To understand the components of the social network.
- To execute the models and visualize the social network using tools.
- To learn the mining of communities in the social network.
- To study the evolution of social networks.
- To know the applications in real time systems.

UNIT I INTRODUCTION 9

The Semantic Web: Limitations of current web – Development of semantic web – Emergence of the social web; Statistical properties of social networks; Social Network Analysis: Development of social network analysis – Key concepts and measures in network analysis – Discussion; Blogs and online communities – Web-based networks.

UNIT II MODELING AND VISUALIZATION 9

Modelling and aggregating social network data: Aggregating and reasoning with social network data – Advanced representations; Visualization of Social Networks: Centrality – Clustering – Node-Edge diagrams; Visualizing online social networks: Node-Link Diagrams – Matrix-based representations.

UNIT III MINING COMMUNITIES 9

Random walks in social networks and their applications: – Random Walks on graphs – Algorithms – Applications; Detecting Communities in Social Networks: Core methods – Emerging fields and problems; Node classification in social networks.

UNIT IV EVOLUTION 9

Evolution in Social Networks: Framework – Tracing smoothly evolving communities; Models and Algorithms for Social Influence Analysis: Influence related statistics – Social similarity and influence – Influence maximization in viral marketing; Algorithms and Systems for Expert Location in Social Networks: Expert location without graph constraints – Score propagation – Expert team formation; Link Prediction in Social Networks: Feature based link prediction – Bayesian probabilistic models – Probabilistic relational models.

UNIT V APPLICATIONS 9

A learning-based approach for real time emotion classification of tweets; A new linguistic approach to assess the opinion of users in social network environments; Explaining scientific and technical emergence forecasting; Social network analysis for biometric template protection.

TOTAL PERIODS: 45

OUTCOMES

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

- Understand the components of the social network (K2).
- Have working knowledge on the models and visualize the social network using tools(K3).

- Gain knowledge on mining of communities in the social network (K2).
- Understand the evolution of the social network (K2).
- Work on the applications in real time systems(K3).

REFERENCE BOOKS

1. Peter Mika, Social Networks and the Semantic Web, Springer, 1st edition, 2007.
2. Borko Furht,Handbook of Social Network Technologies and Applications, Springer, 1st edition, 2011.
3. Charu C. Aggarwal, Social Network Data Analytics, Springer, 2014.
4. Przemyslaw Kazienko, Nitesh Chawla,Applications of Social Media and Social Network Analysis, Springer, 2015.
5. Giles, Mark Smith, John Yen, Advances in Social Network Mining and Analysis, Springer, 2010.
6. Guandong Xu , Yanchun Zhang and Lin Li, Web Mining and Social Networking – Techniques and applications, Springer, 1st edition, 2012.
7. Ajith Abraham, Aboul Ella Hassanien, Vaclav Snasel, Computational Social Network Analysis: Trends, Tools and Research Advances, Springer, 2012.

Course Code	Course Title	Category	L	T	P	C
PCP1379	DEEP LEARNING	PE	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 the advanced deep learning models
- To learn deep learning algorithms and their applications to solve real world problems.

UNIT I DEEP NETWORK 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: Stride – Tiled – Transposed and dilated convolutions; CNN Learning: Nonlinearity Functions – Loss Functions – Regularization – Optimizers – Gradient Computation – CNN through Visualization; CNN Architectures: LeNet – AlexNet – VGGnet – ResNet – ResNeXt.

UNIT III RECURRENT NEURAL NETWORKS 9

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 – Gated RNN.

UNIT IV AUTOENCODERS AND GENERATIVE MODELS 9

Autoencoders: Undercomplete autoencoders – Regularized autoencoders – Stochastic encoders and decoders – Learning with autoencoders; Representation Learning: Unsupervised pretraining – Transfer learning and domain adaptation; Deep Generative Models: Variational autoencoders – Generative adversarial networks.

UNIT V DEEP LEARNING WITH TENSORFLOW 9

TensorFlow: Basics – Optimizers – XOR implementation – Multi-class classification; CNN: Components of CNN – Backpropagation – Dropout layer – Digit recognition – Solving real-world problems; NLP Using RNN: Word2Vec – Network prediction and Sentence completion.

TOTAL PERIODS: 45

OUTCOMES

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

- Explain basics in deep neural networks (K2)
- Apply Convolution Neural Network for real-world problems in image processing (K3)
- Apply Recurrent Neural Network and its variants for text analysis (K3)
- Explain the concepts in autoencoders and generative models (K2)

REFERENCE BOOKS

1. Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, MIT Press, 2016.
2. Salman Khan, Hossein Rahmani, Syed Afaq Ali Shah, Mohammed Bennamoun, A Guide to Convolutional Neural Networks for Computer Vision, Synthesis Lectures on Computer Vision, Morgan & Claypool publishers, 2018.
3. Yoav Goldberg, Neural Network Methods for Natural Language Processing, Synthesis Lectures on Human Language Technologies, Morgan & Claypool publishers, 2017.
4. Santanu Pattanayak, Pro Deep Learning with TensorFlow: A Mathematical Approach to Advanced Artificial Intelligence in Python, Apress, 2017.

Course Code	Course Title	Category	L	T	P	C
PCP1327	**FORMAL VERIFICATION**	**PE**	**3**	**0**	**0**	**3**

OBJECTIVES

- To understand the need of logic 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 symbolic models checking for LTL and CTL.
- To learn Bounded model checking for LTL.

UNIT I FOUNDATIONS 9

Foundations; Propositional Logic: Syntax – Semantics – Resolution; Predicate Logic: Syntax – Semantics – Resolution; LTL; Syntax – Semantics – Specifications in LTL; CTL: Syntax – Semantics – Specifications in CTL; CTL* and the expressive powers of LTL and CTL

UNIT II HOARE LOGIC AND PROGRAM CORRECTNESS 9

Foundations; 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

Foundations; 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 BOUNDED MODEL CHECKING 9

Foundations; SAT/SMT Solvers; Bounded Semantics of LTL; Propositional Encodings of LTL; Completeness; Induction; Interpolation; Completeness with Interpolation; Invariant Strengthening; Bounded Model Checking for Hybrid Automata.

TOTAL PERIODS: 45

OUTCOMES

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

- Prove partial correctness of simple programs using Hoare logic. (K3)
- Write formal specifications in CTL and LTL. (K3)
- Specify and verify simple systems using NuSMV. (K3)
- Write programs using SAT/SMT solvers. (K3)

REFERENCE BOOKS

1. Edmund Clarke, Orna Grumberg, Doron Peled, Model Checking, The MIT Press, 1999. (Units III and IV)
2. Huth M., Ryan M., Logic in Computer Science–Modeling and Reasoning about systems, Cambridge University Press, Second Edition, 2004.(Units I,II and III)
3. Armin Biere Bounded Model Checking, Handbook of Satisfiability, IOS Press, 2009. (Unit V)
4. Baier C., Katoen J., Principles of Model checking, The MIT Press, 2008.

Course Code	Course Title	Category	L	T	P	C
PCP1328	STORAGE AND SERVER SECURITY	PE	3	0	0	3

OBJECTIVES

- To understand the storage systems architecture and available technologies.
- To learn and identify suitable storage networking technologies.
- To identify the areas to provide security in storage infrastructure.
- To learn the security aspects of data center.
- To learn the security protocols and technologies with respect to infrastructure.

UNIT I STORAGE SYSTEMS ARCHITECTURE 9

Introduction to Information Storage and Management : Evolution of Storage Technology and Architecture – Data Center Infrastructure – Key Challenges in Managing Information – Information Lifecycle – Components of a storage system environment; Data Protection: Implementation of RAID – RAID Array components; Different RAID levels and their suitability for different application environments: RAID 0, RAID 1, RAID 3, RAID 4, RAID 5, RAID 0+1, RAID 1+0, RAID 6 – RAID impact on disk performance; Compare and contrast integrated and modular storage systems; High-level architecture and Working of an Intelligent storage system.

UNIT II STORAGE NETWORKING TECHNOLOGIES 9

Direct Attached Storage (DAS); Network Attached Storage (NAS); Storage Area Network (SAN); Internet Protocol SAN (IP-SAN); Content Addressed Storage(CAS).

UNIT III SECURING THE STORAGE INFRASTRUCTURE AND SECURITY IMPLEMENTATIONS 9

Storage Security Framework; Risk Triad: Assets – Threats – Vulnerability; Storage Security Domains: Securing the Application Access Domain – Securing the Management Access Domain – Securing Backup, Recovery, and Archiving (BURA); SAN: Security Architecture – Security Mechanisms; NAS: File Sharing - Windows & Linux – Kerberos; Network-Layer Firewalls; IP SAN: Authentication Protocol – iSNS discovery domain.

UNIT IV DATA CENTER SECURITY OVERVIEW 9

Data Center Design: Types of Server Farms and Data Center; 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 AND TECHNOLOGIES 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 storage systems architecture (K2)
- Choose an appropriate storage networking technology for the given specification (K3)
- Identify Security Implementations in Storage Networking (K3)
- Explain the security aspects in designing a datacenter (K2)
- Design the security protocols for the infrastructure (K3)

REFERENCE BOOKS

1. EMC Corporation, Information Storage and Management: Storing, Managing, and Protecting Digital Information in Classic, Virtualized, and Cloud Environments, EMC Education Services, John Wiley and Sons, 2nd Edition, 2012.
2. Mauricio Arregoces, Maurizio Portolani, Data Center Fundamentals, Cisco Press, 2003.
3. Robert Spalding, Storage Networks: The Complete Reference, McGraw Hill, 2017.
4. John Chirillo and Scott Blaul, Storage Security: Protecting SANs, NAS and DAS, John Wiley & Sons, 2003.
5. Thejendra B. S, Disaster Recovery and Business Continuity, IT Governance Publishing, 3rd Edition, 2014.

Course Code	Course Title	Category	L	T	P	C
*******	**SCIENTIFIC PROGRAMMING USING PYTHON**	**OE**	**2**	**0**	**2**	**3**

OBJECTIVES

- To learn fundamental concepts in Python programming such as variables, flow control structures and to work with lists and sequence data.
- To write Python functions to facilitate code reuse.
- To introduce array computing and to work with dictionaries and strings.
- To introduce mathematical modeling and to work with discrete calculus, differentiation and integration.

UNIT I PYTHON FUNDAMENTALS 9

Introduction – Variables – Objects – Arithmetic operators – Importing a module – Interactive computing – Python shell – Type conversion – Symbolic computing; Loops and lists: While loop – For loops – Lists – List comprehension – Nested lists – Extracting sublists – Tuples.

UNIT II FUNCTIONS AND BRANCHING, FILE HANDLING 9

Functions and branching: Functions – Local and global variables – Multiple arguments – Multiple return values – Functions as arguments – Branching: If else blocks – Inline if tests. User input: Reading keyboard input – Reading from the command line – Reading data from file – Writing data to file.

UNIT III ARRAYS, DICTIONARIES 8

Array computing and curve plotting: Vectors – Basics of numerical Python arrays – vectorization – Curve plotting – Matrices and arrays. Dictionaries: Making dictionaries – Dictionary operations.

UNIT IV STRINGS, CLASSES 10

Strings: Operations on strings – Reading and writing spreadsheet files; Introduction to classes: Simple function classes – Representing a function as a class – Special methods – Adding objects – Static methods and attributes.

UNIT V SEQUENCES AND DIFFERENCE EQUATIONS 9

Mathematical models based on difference equations; Introduction to discrete calculus: Discrete functions – Differentiation becomes finite differences – Integration becomes summation – Taylor series; Introduction to differential equations – Exponential growth – Logistic growth.

Suggested Experiments

1. Given the list L find the min, max and median value. Create a list with only the elements at even indexes and re-compute min, max and median values.
2. Write a script to build a new list containing the square roots of all the numbers in the given list. Implement it with list comprehension.
3. Consider the matrix M and the vector v. What is the matrix-vector product $M * v$ and square of M?

4. Write a function $sin(M)$ that computes it with sin series.
5. Write a function to find the matrix rank by computing the number of numerical non-zero singular values.
6. Write a program that counts the number of lines and words in a file, printing the resulting counts to the terminal.
7. Write a method *simplify* for the class *Rational Number*. This method should return the simplified fraction as a tuple.
8. Write a program to count the characters in a string and store it as a dictionary, using the characters as the keys and the counts as values.
9. Create a string containing at least five words and store it in a variable.
    1) Convert the string to a list of words using the split method.
    2) Sort the list into reverse alphabetical order using some of the list methods.
10. Define a class called Rectangle with width and height as attributes. Create a member attribute corner which itself is an object of another class Point with x and y as attributes. Add an equals method to the Rectangle class and perform a deep equality check. Add a method called move to Rectangle class to perform a translation.
11. Write a program to randomly generate a list with 5 numbers, which are divisible by 5 and 7 , between 1 and 1000 inclusive. Use *random.sample()* to generate a list of random values.
12. Solve the ODE problem: $u' = 2u - 1$, $u(0) = 2$, $t\ in\ [0, 6]$ using the Forward Euler method. Choose $\Delta t = 0.25$. Plot the numerical solution together with the exact solution.

$$u(t) = 1/2 + 3/2\, e^{2t}.$$

13. Write a program for Taylor series for e^x with n terms for various values of h and print the error of the approximations.

TOTAL PERIODS: 60

OUTCOMES

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

- Write Python programs using decision and repetition structures in program design. (K3)
- Demonstrate a basic knowledge of numerical computing using functions, error handling and file handling. (K3)
- Develop applications to demonstrate the understanding of arrays dictionaries and strings. (K3)
- Write code in Python to solve problems requiring the discrete calculus, Differentiation, Integration and scientific simulations. (K3)

REFERENCE BOOKS

- Hans Petter Langtangen, A Primer on Scientific Programming with Python, Springer, 4th Edition, 2016.

Course Code	Course Title	Category	L	T	P	C
*******	INTRODUCTION TO MACHINE LEARNING	OE	2	0	2	3

OBJECTIVES

- To have a basic knowledge of the concepts and tools of machine learning.
- To learn the data pre-processing methods and be able to apply on dataset
- To understand the working of supervised and unsupervised algorithms.
- To learn the evaluation methods and apply them for validation

UNIT I INTRODUCTION TO MACHINE LEARNING 9

Why Machine Learning: Problems Machine Learning Can Solve – Task and Data; scikit-learn; Essential Libraries and Tools: Jupyter Notebook – NumPy – SciPy – matplotlib – pandas – mglearn; Classifying Iris Species: Data – Measuring Success – Training and Testing Data – Building Model: k-Nearest Neighbors – Making Predictions – Evaluating the Model.

UNIT II DATA PRE-PROCESSING 9

Preprocessing and Scaling: Different Kinds of Preprocessing – Applying Data Transformations – Scaling Training and Test Data; Categorical Variables: One- Hot-Encoding – Numbers to Encode Categorical; Binning, Discretization, Linear Models, and Trees; Interactions and Polynomials; Univariate Nonlinear Transformations; Automatic Feature Selection: Univariate Statistics – Model- Based Feature Selection – Iterative Feature Selection.

UNIT III SUPERVISED LEARNING 9

Classification and Regression; Generalization, Overfitting, and Underfitting; Relation of Model Complexity to Dataset Size; Supervised Machine Learning Algorithms: Sample Datasets – k-Nearest Neighbors – Linear Models – Naïve Bayes Classifiers – Decision Trees – Ensembles of Decision Trees – Bagging – Random Forests – Boosting – Neural Networks (Deep Learning).

UNIT IV UNSUPERVISED LEARNING 9

Types of Unsupervised Learning; Challenges; Dimensionality Reduction: Principal Component Analysis (PCA); Clustering – k-Means Clustering Agglomerative Clustering – Comparing and Evaluating Clustering Algorithms.

UNIT V MODEL EVALUATION AND IMPROVEMENT 9

Cross-Validation: Cross-Validation in scikit-learn – Benefits of Cross-Validation – Stratified k-Fold Cross-Validation and Other Strategies; Grid Search: Simple Grid Search – The Danger of Overfitting the Parameters and the Validation Set – Grid Search with Cross-Validation; Evaluation Metrics and Scoring: Keep the End Goal in Mind – Metrics for Binary Classification – Metrics for Multiclass Classification – Regression Metrics – Using Evaluation Metrics in Model Selection.

Suggestive Experiments (Python - NumPy, Scikit-learn, Matplotlib)

- Perceptron and Linear Regression
- Multi-layer Perceptron
- Support Vector Machine

- Decision Tree algorithm
- k-Nearest Neighbor algorithm
- K-means clustering

TOTAL PERIODS: 60

OUTCOMES

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

- Understand the basic concepts of machine learning (K2).
- Apply data preprocessing and feature selection for datasets (K4).
- Apply supervised and unsupervised techniques for different applications (K4).
- Evaluate and analyze the models using appropriate metrics (K4).

REFERENCE BOOKS

1. Andreas C. Muller, Sarah Guido, Introduction to Machine Learning with Python, O'Reilly Media, 2016.
2. Aurelien Geron, Hands-On Machine Learning with Scikit-Learn and TensorFlow, O'Reilly Media, 2016
3. Sebastian Raschka, Vahid Mirjalili, Python Machine Learning, Packt Publishing, Second Edition, 2017.
4. Stephen Marsland, Machine Learning - An Algorithmic Perspective, Chapman and Hall/CRC Machine Learning and Pattern Recognition Series, Second Edition, 2014.

Course Code	Course Title	Category	L	T	P	C
*******	INTRODUCTION TO BIG DATA ANALYTICS	OE	2	0	2	3

OBJECTIVES

- To understand the competitive advantages of data analytics
- To understand the data frameworks
- To learn data analysis methods
- To gain knowledge on Hadoop related tools such as HBase, and Hive for data analytics

UNIT I INTRODUCTION TO BIG DATA **8**

Big Data: Definition – Characteristic features – Big data applications – Big data vs Traditional data – Risks of big data – Structure of big data; Web data; Evolution of analytic scalability; Modern Data Analytic Tools.

UNIT II HADOOP FRAMEWORK **10**

Distributed File Systems: Large-scale file system organization – HDFS concepts – MapReduce execution – Algorithms using MapReduce.

UNIT III DATA ANALYSIS **9**

Statistical Methods: Regression modelling – Multivariate analysis; Classification: SVM & Kernel methods – Decision Trees; Linear Classifiers.

UNIT IV NoSQL **9**

Introduction to NoSQL – Characteristics of NoSQL – NoSQL Storage Types – Advantages and Drawbacks.

UNIT V BIG DATA FRAMEWORKS **9**

MongoDB – Aggregate data models – HBase: Data model and implementations – HBase clients; Hive: Data types and file formats – HiveQL data definition – HiveQL data manipulation – HiveQL queries.

Suggestive Experiments

Hadoop

- Applications using Map-Reduce programming (Examples: word count / frequency programs / matrix multiplication)

R

- Linear and logistic Regression (Loan prediction using Credit approval dataset, Sales prediction using Bigmart dataset)
- SVM / Decision tree classification techniques (Flower type classification based on available attributes using Iris dataset, Passengers survival classification using titanic dataset)
- Clustering (Document categorization by multiclass techniques)
- Visualize data using any plotting framework.

Database

- Application that stores data in HBase / MongoDB (Sentiment analysis using twitter dataset)

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)
- Solve applications using statistical and data analytic methods (K3)
- Develop applications using Hadoop related tools(K4)
- Use database frameworks like MongoDB, Hive and HBase for data analysis(K3)

REFERENCE BOOKS

1. Bill Franks, Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics, John Wiley & sons, 2012.
2. Michael Berthold, David J. Hand, Intelligent Data Analysis, Springer, 2007.
3. Tom White, Hadoop: The Definitive Guide - Storage and Analysis at Internet Scale, O’Reilly, 4th Edition, 2015.
4. Gaurav Vaish, Getting Started with NoSQL, Packt Publishing Ltd, 2013.
5. Capriolo E., Wampler D., Rutherglen J., Programming Hive, O’Reilly, 2012.
6. Lars George, HBase: The Definitive Guide, O’Reilly, 2011.
7. Kristina Chodorow, MongoDB: The Definitive Guide – Powerful and Scalable Data Storage, O’Reilly, 2nd Edition, 2013.