N.C.College of Engineering

Israna-132107 (Panipat)

Scheme and Syllabus

w.e.f 2015-16 Session

Computer Science & Engineering

Fourth Year (7th & 8th Semester)

MARKS DISTRIBUTION
(ACCORDING TO AUTONOMY)

FOR ALL THEORY COURSES: -

1. On Semester Evaluation of all theory courses
   Total: 100 Marks

Distribution	
I. Mid Semester Examination	20 Marks
II. Mid Semester Examination	20 Marks
III. Mid Semester Examination	20 Marks
Continuous Evaluation Test (CET)	20 Marks
Attendance	20 Marks
Teacher's Assessment	20 Marks

} { 40 Marks } Best two will be included

2) End Semester (Final Examination) of all theory courses

   Total: 100 Marks

3) Total of On Semester + End Semester Evaluation is of 200 Marks

4) To pass a theory course, the student should obtain

   Minimum: - 80 Marks out of 200.

Criterion for passing and failing in the theory courses: -

a) The students will have to obtain 35% Marks in theory and 80 Marks in aggregate of On Semester and End Semester Evaluation for passing. If the above passing criterion is not fulfilled, the student will be awarded "Reappear".

b) On Semester Marks will not be changed. Only the theory marks will be modified as obtained in "Reappear".

c) If the attendance in a course is below 75%, the student will not be permitted to appear in the Final Examination.

FOR ALL PRACTICAL (LABORATORY) COURSES: -

I) On Semester Evaluation of all Practical (Laboratory) Courses

Total: 120 Marks.

Distribution

Attendance	60 Marks
Record of Practicals/ Experiments	30 Marks
Teacher's Assessment	30 Marks

II) End Semester Evaluation (Final Lab Examination + Oral Test or Viva Voce)

Total: 80 Marks

III) Total of On Semester Evaluation (Final Lab Examination) + End Semester Evaluation is of 200 Marks.

IV) To pass a lab course, the student should obtain

Minimum: 80 Marks out of 200.

Criterion for passing and failing in the lab course is just like the theory course.

CALCULATION OF SEMESTER GRADE POINT AVERAGE: -

Semester grade point average (SGPA) is the weighted average of the grade for the subjects registered **in a Semester** and is computed as follows:

$$SGPA = \frac{\sum_i C_i \times G_i}{\sum_i C_i}$$

C_i denotes the Credits (or Units) assigned to the ith subject and G_i denotes the Grade Point Equivalent to the Letter Grade obtained for the ith subject.

Cumulative Grade Point Average (CGPA) is the weighted average of the grades of the subjects for the registered in the semester.

N. C. COLLEGE OF ENGINEERING, ISRANA

SCHEME OF STUDIES AND EXAMINATION

B. Tech. – Computer Science & Engineering

4th Year (Semester–VII) 2015-19

Sr. No.	Course no.	Subjects		Teaching Schedule			Contact Hours	Credits
			BOS	L	T	P		
1	CSE-471	Compiler Design	CSE	3	1	-	4	4
2	CSE-472	Advance Computer Architecture	CSE	3	1	-	4	4
3	CSE-473	Artificial Intelligence	CSE	3	1	-	4	4
4	CSE-474	Mobile Applications Development	CSE	3	1	-	4	4
5	SSAA-477	Soft Skills and Analytical Ability- II	CSE	1	-	2	3	2
6	CSE-47P1	Minor Project	CSE				6	6
7	CSE-47P2	Seminar	CSE	-	-	2	2	1
8	CSE-47P3	PROLOG Lab	CSE	-	-	2	2	1
9		Elective-I	CSE	3	1	-	4	4
		Total		16	5	6	**33**	**30**

Elective-I

1. Software Project management (CSE-47E1)
2. Open source technologies (CSE-47E2)
3. Image Processing (CSE-47E3)
4. Statistical Model for Computer Science (CSE-47E4)
5. Object Oriented Software Design using UML (CSE-47E5)
6. **Mathematics-IV** **(MATH-OE1)**

7th Semester (Computer Science & Engineering)

COMPILER DESIGN

NC-CSE-471

L	T	
3	**1**	**On Semester Evaluation: 100 Marks** **End Semester Evaluation: 100 Marks**

Note: - 1. There are NINE questions in a set of question-paper. All questions carry equal marks.

2. Attempt five questions in all. FIRST question is compulsory which covers the whole syllabus. Attempt ONE question from each of the other four Units.

UNIT-I

Assemblers, linkers, loaders, compilers and translators, the structure of compiler, different states in the construction of a compiler, Design of lexical analyzer, Basic Parsing Techniques, Parsers, shift-reduce parsing, operator-precedence parsing, top-down parsing predictive parsers, L.R. Parsers, the canonical collection of L.R (O) items, construction of SLR parsing tables, construction canonical L.R. Parsing tables, Constructing LALR parsing tables implementation of L.R. Parsing tables.

UNIT-II

Syntax-Directed Translation: Syntax-directed translation schemes, implementation of syntax directed translators, intermediate code, postfix notation, parse trees and syntax trees, three address code, quadruples, and triples, translation of assignment statements. Boolean expressions, control statements.

Symbol Labels: The contents of a symbol table data structures for symbol tables representing scope information.

UNIT-II

Run time storage Administration: Implementation of a simple stack allocation scheme, implementation of block structured languages, storage allocation in block structured languages.

Error Detection and Recovery: Error, Lexical-phase errors, syntactic-phase errors, semantic errors.

UNIT-IV

Code Optimization: The Principle sources of optimization, loop optimization, the DAG representation of basic blocks, value number and algebraic laws, global dataflow analysis.

Code Generation: Object programs, problems in code generation, a machine model, a single code generator, register allocation and assignment, code generation from DAGs, peephole optimization.

Books:

1. Aho A.V. and ullaman J.D.Principles of Compiler Design, Addison Wesley
2. Donovan, j, System Programming, TMH
3. D.M. Dhamdhere: Compiler Construction- Principles and practice Mc Milan India

David Grics : Compiler Construction for digital computer

7th Semester (Computer Science & Engineering)

ADVANCED COMPUTER ARCHITECTURE

CSE-472

L	**T**	
3	**1**	**On Semester Evaluation: 100 Marks** **End Semester Evaluation: 100 Marks**

Note: - 1. There are NINE questions in a set of question-paper. All questions carry equal marks.

2. Attempt five questions in all. FIRST question is compulsory which covers the whole syllabus. Attempt ONE question from each of the other four Units.

UNIT-I

INSTRUCTION LEVEL PARALLELISM

ILP – Concepts and challenges – Hardware and software approaches – Dynamic scheduling – Speculation - Compiler techniques for exposing ILP – Branch prediction.

MULTIPLE ISSUE PROCESSORS

VLIW & EPIC – Advanced compiler support – Hardware support for exposing parallelism – Hardware versus software speculation mechanisms – IA 64 and Itanium processors – Limits on ILP.

UNIT-II

MULTIPROCESSORS AND THREAD LEVEL PARALLELISM

Symmetric and distributed shared memory architectures – Performance issues – Synchronization – Models of memory consistency – Introduction to Multithreading.

UNIT-III

MEMORY AND I/O

Cache performance – Reducing cache miss penalty and miss rate – Reducing hit time – Main memory and performance – Memory technology. Types of storage devices – Buses – RAID – Reliability, availability and dependability – I/O performance measures – Designing an I/O system.

UNIT-IV

MULTI-CORE ARCHITECTURES

Software and hardware multithreading – SMT and CMP architectures – Design issues – Case studies – Intel Multi-core architecture – SUN CMP architecture – heterogenous multi-core Processors – case study: IBM Cell Processor.

TEXT BOOK:

1. John L. Hennessey and David A. Patterson, " Computer architecture – A quantitative approach", Morgan Kaufmann / Elsevier Publishers, 4th. edition, 2007.

REFERENCES:

1. David E. Culler, Jaswinder Pal Singh, "Parallel computing architecture : A hardware/software approach" , Morgan Kaufmann /Elsevier Publishers, 1999.

2. Kai Hwang and Zhi.Wei Xu, "Scalable Parallel Computing", Tata McGraw Hill, New Delhi, 2003.

7th Semester (Computer Science & Engineering)

ARTIFICIAL INTELLIGENCE

CSE-473

L	**T**	
3	**1**	**On Semester Evaluation: 100 Marks** **End Semester Evaluation: 100 Marks**

Note: - 1. There are NINE questions in a set of question-paper. All questions carry equal marks.

2. Attempt five questions in all. FIRST question is compulsory which covers the whole syllabus. Attempt ONE question from each of the other four Units.

UNIT-I

Introduction: Definition of Artificial Intelligence (AI), Evolution of Computing History of AI, Classical Romantic and modern period, Subject area, Architecture of AI machines, logic family, classification of logic.

Production System: Production rules, the working memory, Recognize-act cycle, conflict resolution by meta rules, Architecture of production system.

UNIT-II

Propositional Logic: Proposition, tautologies, Theorem proving, Semantic method of theorem proving, forward chaining, backward chaining standard theorems, method of substitution, Theorem proving using Wang's algorithm.

Predicate Logic: Alphabet of first order logic(FOL), predicate, well formed formula, clause form, algorithm for writing sentence into clause form, Unification of predicates, unification algorithm, resolution Robinson's interface rule, Scene interpretation using predicate logic.

UNIT-III

Default and Non monotonic Logic: Axiomatic theory, Monotonicity, non-atomic reasoning using McDermott's NML-I, problems with NML-I, reasoning with NML-II, Case study of Truth Maintenance System(TMS), neural network fundamentals.

Imprecision and Uncertainty: Definition, Probabilistic techniques, Certainty factor based reasoning, conditional probability, Medical diagnosis problem, Baye's Theorem and its limitations, Bayesian belief network, propagation of belief, Dumpster-Shafer theory of uncertainty management, belief interval, Fuzzy relation, inverse Fuzzy relations, Fuzzy post inverse, Fuzzy Inversion.

UNIT-IV

Intelligent Search Techniques: Heuristic function, AND-OR graph, OR Graph, Heuristic search, A* algorithm and examples.

Logic Programming with Prolog: Logic program, Horn clause, program for scene interpretation, unification of goals, SLD resolution, SLD tree, flow of satisfaction, controlling back track using CUT, Command use of CUT, implementation of backtracking using stack, risk of using cuts, fail predicate, application of cut-fail combination, replacing cut-fail by not.

Books:

1. A.Konar: Artificial Intelligence and Soft Computing-Behavioral and Cognitive Modeling of Human Brain, CRC Press, USA.
2. E. Charniak and D.McDermott: Introduction to Artificial Intelligence, Addison Welley Longman.
3. Rich and Knight: Artificial Intelligence, 2/e 1992

7th Semester (Computer Science & Engineering)

MOBILE APPLICATIONS DEVELOPMENT

CSE-474

L	T	
3	**1**	**End Semester Evaluation: 100 Marks**

On Semester Evaluation: 100 Marks

Note: - 1. There are NINE questions in a set of question-paper. All questions carry equal marks.

2. Attempt five questions in all. FIRST question is compulsory which covers the whole syllabus. Attempt ONE question from each of the other four Units.

UNIT-I

Getting Started with Mobility: Mobile operating system, Operating system structure , Constraints and Restrictions, Hardware configuration with mobile operating system, Mobile apps development, Overview of Android platform, setting up the mobile app development environment along with an emulator, a case study on Mobile app development.

UNIT-II

Building blocks of mobile apps: App user interface designing- mobile UI resources(Layout,UI elements, Draw-able, Menu), Activity states and life cycle, interaction amongst activities.

App functionality beyond user interface- Threads, Async task, Services- states and life cycle, Notifications, Broadcast receivers, Telephony and SMS APIs

Native data handling- on device file I/O, shared preferences, mobile databases such as SQLite, and enterprise data access(via Internet/Intranet)

UNIT-III

Sprucing up mobile apps: Graphics and animation - custom views, canvas, animation APIs, multimedia - audio/video playback and record, location awareness, and native hardware access (sensors such as accelerometer and gryoscope)

UNIT-IV

Testing mobile apps and taking to Market and other Mobile Operating Systems: Debugging mobile apps, White box testing, Black box testing, and test automation of mobile apps, JUnit for Android, Robtium, MonkeyTalk

Versioning, signing and packaging mobile apps, distributing apps on market place

Introduction to IOS and Window phones their Architecture , memory management , communication protocols application development method , deployment and case study.

Text Books:

1. Learn Java for Android Development by Jeff Friesen, Apress

2. Andrid Application Development, Rick Rogers by John Lambardo, Zignard Mcdnicks, G Blake Maike, O"Reilley Media Inc

References:

1. Andriod Apllication Development All-in-one for Dummies, Bary Burd by Jon Valley & Sons Inc.

2. Sam Teach Yourself Andriod Application Development in 24 hours by Lauren Darcev, Shane Conder SAMS.

7th Semester (Computer Science & Engineering)

SOFT SKILLS AND ANALYTICAL ABILIY-II

NC-ESC-401

L	T	P	
1	0	2	**On Semester Evaluation: 100 Marks**
			End Semester Evaluation: 100 Marks

UNIT -I

Quantative Aptitude-

1. Probability
2. Permutations & Combinations
3. Area and Volume
4. Calendar, Clocks
5. Alligation & Mixture
6. Time, speed and distance/ upstream and downstream /Train
7. Pipe and Cistern
8. Partnership.

UNIT –II

Qualitative Aptitude-

1. Logical Venn Diagram
2. Data Sufficiency
3. Assertion & reason,
4. Statements- Arguments, Assumptions, Course of action, Conclusion.
5. Cause & effects

UNIT -III

1. Business communication skills

- English language enhancement – 1 lecture
- Art of communication- 2 lectures

2. Intrapersonal and interpersonal relationship skills Interpersonal skills

- Intrapersonal skills
- Effective team player

3. Campus to company

- Corporate dressing
- Corporate grooming
- Business etiquette
- Communication media etiquettes

4. Group discussions, Interviews, Presentations

5. Entrepreneur skills development Goal setting

- Entrepreneur skills- awareness and development

Syllabus for Practical Classes

1. Group discussions: topics to be given in the class
2. Mock Interviews
3. Presentations

7th Semester (Computer Science & Engineering)

SOFTWARE PROJECT MANAGEMENT

CSE-47E1

L	T	
3	**1**	**On Semester Evaluation: 100 Marks** **End Semester Evaluation: 100 Marks**

Note: - 1. There are NINE questions in a set of question-paper. All questions carry equal marks.

2. Attempt five questions in all. FIRST question is compulsory which covers the whole syllabus. Attempt ONE question from each of the other four Units.

UNIT-I

Conventional Software Management: Evolution of software economics, Improving software economics: reducing product size, software processes, team effectiveness, automation through, p Software environments, Principles of modem software management.

UNIT-II

Software Management Process: Framework,: Life cycle phases- inception, elaboration, construction and training phase. Artifacts of the process- the artifact sets, management artifacts, engineering artifacts, and pragmatics artifacts, Model based software architectures, Workflows of the process, Checkpoints of the process.

UNIT-III

Software Management Disciplines: Iterative process planning, Project organizations and responsibilities, Process automation, Project control and process instrumentation core metrics, management indicators, life cycle expectations, Process discriminates.

UNIT-IV

Project Quality Management: importance and role of software quality, quality panning, quality control, quality assurance, quality standards, project risk management, common sources of risks and on IT projects, risk identification, risk quantification, risk resource development and control.

Books:

1. Software Project Management, Walker Royee, Addison Wesley, 1998
2. Project management 2/e, Maylor.
3. Managing the Software Process, Humphrey.
4. Managing Global Software Projects. Ramesh, TMfH, 2001

7th Semester (Computer Science & Engineering)

OPEN SOURCE TECHNOLOGIES

CSE-47E2

L	**T**	
3	**1**	**On Semester Evaluation: 100 Marks** **End Semester Evaluation: 100 Marks**

Note: - 1. There are NINE questions in a set of question-paper. All questions carry equal marks.

2. Attempt five questions in all. FIRST question is compulsory which covers the whole syllabus. Attempt ONE question from each of the other four Units.

UNIT-1

Introduction: open Source, Free Software, Free Software vs. Open Source software, Public Domain Software, FOSS does not mean any cost. History: BSD, The Free Software Foundation and the GNU Project, Open source development models.

Principles and Open Source Methodology, Initiatives, Philosophy: Software Freedom, Problems with traditional commercial software, Internationalization. Open source vs. closed source

UNIT-II

Licenses and Patents: What Is A License, Important FOSS Licenses (Apache, BSD, GPL, LGPL), copyrights and copylefts, Patents Economics of FOSS : Zero Marginal Cost, Income-generation opportunities, Starting and Maintaining an Open Source Project, Open Source Hardware, Open Source Design, Open source Teaching and Open source media.

Open source government, Open source ethics. Social and financial impacts of open source technology, Open Standards: National Information Standards Organization (NISO), The Digital Library Federation (DLF).

UNIT-III

Linux: Overview of Linux, Overview of UNIX and LINUX Architectures, Advantages of Linux, Linux File System, Hardware requirements for Linux, System installation**.**

Linux Commands, Users and groups; file system and directories. Software Management: RPM, Debian, Installing software.

Processes: system processes, internal and external commands, background process, premature termination of process, process priorities, process scheduling-(at, batch), nohup command

UNIT-IV

Shell in Linux: Available shells under Linux (viz. Bash, TCSH, Korn or so on), shell scripts: shell variables, environmental variables, purpose of shell scripts, writing, storing and executing scripts, Filter- The grep family.

Kernel tasks; managing kernel modules at runtime; kernel configuration and compilation boot loaders GRUB and LILO.

Remote procedure call (R P C) services, NFS server and client sides, NFS installation & configuration; and statistic mount and auto mount

Text Books:

1. Open source: technology and policy, Fadi P. Deek, James A. McHugh, Cambridge University Press,
2. Open Source Technology, Kailash Vadera, Bhavyesh Gandhi, University Science Press, First Edition.
3. Complete Reference, Red Hat Linux-Richard L. Peterson – TMH
4. Unix Concepts And Applications, Das, Sumitabha, Tata McGraw-Hill Education

References:

1. The success of open source, Steven Weber, copyright © 2004, printed in United States of America.
2. Cox K. "Red Hat Linux Administrator's Guide", PHI, 2001.
3. Yashwanr Kanetkar, "Unix shell programming", BPB Publications.
4. Neil M:HhcIV. Richard Stones. "Beginning Linux Programming", Wrox Press.
5. Website: www.linux.org .

7th Semester (Computer Science & Engineering)

IMAGE PROCESSING

CSE-47E3

L	T	
3	**1**	**On Semester Evaluation: 100 Marks** **End Semester Evaluation: 100 Marks**

Note: - 1. There are NINE questions in a set of question-paper. All questions carry equal marks.

2. Attempt five questions in all. FIRST question is compulsory which covers the whole syllabus. Attempt ONE question from each of the other four Units.

UNIT – I

Image Processing Fourier Transform and Z-Transform, Causality and Stability, Toeplit and Circulate Metrics, orthogonal and unitary Matrices and Kroenker product, Markov Processes KI Transform Mean Square Estimates and Orthogonal Principles.

Image Sampling quantization, Band Limited Image Sampling Versus Replication, Reconstruction of image from samples Sampling Theorem, Sampling Theorem for Random Fields, Sampling Optimal Sampling, Nonrectangular Grid Sampling, Sampling Aperture, Display Aperture/Interpolation Functions, Lang range Interpolation, Moire Effect, Image Quantization Uniform Optimal Quantizer, Properties of Mean Square Quantizer, Commands Design Visual Quantization.

UNIT – II

Image Transforms: Two Dimensional Orthogonal and Unitary Transforms and their properties. One Dimensional and Two Dimensional DFT Cosine and Sine Transforms Iiadmard, slant, IIARR and KI, Transforms and their properties, Approximation to KI Transforms. Image representation by stochastic model, One Dimensional Causal Models, AR and ARMA models, Non Causal Representation Spectral factorization, Image Decomposition

UNIT – III

Image Enhancement and Restoration: Point Operation, Histogram Modeling, Spatial Operations, Transform Operations. MultiSpeciral Image Enhancement. Image Observation Models, Inverse and Wiener Filtering FIR wiener Filters, Filtering using Image Transform Casual Models and recursive filtering Maximum entropy restoration. Extrapolation of band limited signal.

UNIT – IV

Image Analysis and Image Compression: Spatial feature extraction, Edge detection and boundary extraction boundary, region and moment representations structures, Texture, Image Segmentation, Reconstruction from Projections, Pixel Coding, Productive Techniques, Transform Coding Theory, Coding of Image, Coding of two-tone image.

Books:

1. Anil Jain: Digital Image Processing, Prentice Hall 1989, I/E.
2. Gonzalez Woods: Image Processing, Pearson Education India, 2nd Ed.

7th Semester (Computer Science & Engineering)

STATISTICAL MODELS FOR COMPUTER SCIENCE

CSE-47E4

L	T	
3	**1**	**On Semester Evaluation: 100 Marks** **End Semester Evaluation: 100 Marks**

Note: - 1. There are NINE questions in a set of question-paper. All questions carry equal marks.

2. Attempt five questions in all. FIRST question is compulsory which covers the whole syllabus. Attempt ONE question from each of the other four Units.

UNIT-I

Probability Models, Sample Space, Events, their algebra, graphical methods of representing events, Probability Axioms and their applications, Condition probability, Independence of Events, Bayes' Rule and Bernoulli Trials.

UNIT-II

Random variable, and their event space, probability mass function, Distribution functions, some discrete distributions (Bernoulli, Binomial, Geometric, Negative Binomial, Poisson, Hyper Geometric and Uniform), Probability Generating Function, Discrete random vectors. Continuous random variables: some continuous distributions (Exponential, Hyperexponential, Erlang, Gamma, Normal), Functions of random variables, jointly distributed random variables, Expectation, Expectation of functions of more than one random variable, Brief introduction to Conditional pmf: pdf and expectation, Moments and transforms of some distributions (Uniform, Bernoulli, Binomial, Geometric, Poisson, Exponential, Gamma, Normal), Computation of mean time to failure.

UNIT-III

Stochastic Processes, Classification of stochastic processes, the Bernoulli process, The Poisson process, renewal process, renewal model of program behavior.

UNIT-IV

Markov Chains, Computation of n-step transition probabilities, State classification and limiting distributions, Distribution of times between state changes, Irreducible finite chains with aperiodic states, M/G/I queuing system, Discrete parameter Birth-Death processes, Analysis of program execution time. Continuous parameter Markov Chains, Birth-Death process with special cases, Non-Birth-Death Processes.

Books:

1. K.S. Trivedi, Probability, Statistics with Reliability, Queuing and Computer Science Applications, PHI, 2001.
2. J.F. Hayes, Modeling of Computer Communication Networks, Khanna Publishing, Delhi
3. W. Feller, An Introduction to Probability Theory and its applications. 2vols. Wiley Eastern, 1975
4. L. Kleinroek, Queuing Systems, 2vols, John Wiley, 1976.

7th Semester (Computer Science & Engineering)

OBJECT ORIENTED SOFTWARE DESIGN USING UML

CSE-47E5

L	T	
3	**1**	**On Semester Evaluation: 100 Marks**
		End Semester Evaluation: 100 Marks

Note: - 1. There are NINE questions in a set of question-paper. All questions carry equal marks.

2. Attempt five questions in all. FIRST question is compulsory which covers the whole syllabus. Attempt ONE question from each of the other four Units.

UNIT–I

UML:History of UML, Goals of UML, nature & purpose of models, UML views & diagrams– static, design, use case, state machine, activity, interaction deployment, model management, profile; relationships in UML–association, dependency, generalization, realization; UML Extensibility mechanisms– constraints, stereotypes, tagged values.Unified Process (UP): UP structure, phases of UP

UNIT–II

Requirements: Meta Model, Workflow, Functional and Non- functional Requirements; equirement Attributes, Finding Requirements Use Case Modeling: Finding Actors and Use Cases, Use Case Scenario – main flow, branching within a flow, repletion within a flow, Modeling alternative flows; relationships among actors and use cases; use case diagrams

UNIT –III

Analysis: Meta Model , Workflows, Finding Analysis Classes – using noun/verb analysis, RC analysis, using RUP stereotypes - entity, boundary and control; Modeling Classes – Association (role name, multiplicity, navigability, association classes, qualified association) Dependencies (usage, abstraction, permission), class generalization, generalization sets, power types; Analysis Package– nested packages, dependencies, transitivity, package generalization, Architectural analysis, finding analysis packages; Concepts of Patterns & Frameworks Use Case Realization – interaction diagram, sequence diagram; Activity Diagrams.

UNIT–IV

Design: Meta Model, Workflow, design classes – well - formed design classes, inheritance, templates, nested classes, design relationships, aggregation and composition, refining analysis relationships; interfaces and components – provided and required interfaces, interface Realization v/s interface, components, finding interfaces, designing with interfaces; interaction diagram in design, modeling concurrency, active classes, concurrency in sequence diagram, concurrency in communication diagram; state machine - state machine diagrams Implementation: Meta model, workflow, deployment diagram

Text Books:

1. Jim Arlow, Ila Neustadt, "UML 2 and the Unified Process –Practical Object Oriented Analysis and Design", Pearson Education .
2. Bernd Bruegge, Allen H. Dutoit, "Object Oriented Software Engineering using UML", Pearson Education.

Reference Books:

1. Rumbaugh J., Jacobson I., Booch G., "The Unifed Modeling Language Reference Manual", Pearson Education.

2. Blaha M., Rumbaugh J., “Object - Oriented Modeling and Design with U ML”, Pearson Education.
3. Timothy C. Lethbridge, Robert Laganiere, “Object Oriented Software Engineering”, Tata McGraw - Hill.
4. Booch G., Rumbaugh J., Jacobson I., “The Unified Modeling Language User Guide”, Pearson education.
5. Satzinger, Jackson, Burd, “Object - Oriented Analysis & Design with the Unified Process”, Course Technology Inc

7h Semester (Computer Science & Engineering)

PROLOG LAB

CSE-47P3

P	**On Semester Evaluation: 120**
2	**End Semester Evaluation: 80**

List of Experiments:

1. Describe the following family tree as series of PROLOG facts.
   M – Married to
   C – Child of

   Anthony – M – Mary
   | |
   C
   Harry Hazel – M – Tom

   Write queries to answer the following:
   a) Who is the sister of Harry?
   b) Who is the father of Hazel?

2. The following Diagram depicts name of employees and their supervisors.

   Watson

   Johnson John Banker

   Smita Evens

   WAP which contain all the supervisor relationship in diagram and answer the given queries.
   - who is supervisor of evens
   - whom supervisor is Watson

3. Write a PROLOG program that answers about family members and relationships. Include predicates & clauses which define sister, brother, father, mother, Grandchild, grandfather and uncle. The program should be able to answer question such as following.
   Father (X, bob)
   Grandson (X, Y)
   Uncle (bill, Sue)
   Mother (marry, X)

4. Write Prolog Program to Create following database.

P_Name	Age	Hobby	Pet
Ram	15	Football	Dog
Mohan	11	Volleyball	Cat
Sohan	25	Card	Cow
Mahesh	30	Swimming	Dog

Ravindra	11	Football	GOat
Rakesh	25	Volleyball	Cat
Rajeev	15	Swimming	Dog
Raju	30	Swimming	Dog
Jaichand	40	Football	Cow
Vijay	30	Volleyball	Cat

a) Write Query to display P_name and age, P_Name and hobby.

b) Find How many of them are child if age <=15 is child.

c) X will like Y if X & Y are Persons & they are not same in age but they are children & have a common interest then show who likes whom.

5. Write a Prolog code for Monkey Banana Problem.

6. Write a prolog program for insertion sort and bubble sort.

7. a) Write a prolog program to depict the functioning of cuts.
b) Write Prolog program to find the sum of numbers in a list.

8. Write a Program to reverse list of names.

9. Prove that any planar graph cannot be colored with less than 4 colors in prolog.

10. Represent a graph in Prolog and apply Breath first search on it.

Optional Programs:

1. Write a program in prolog to implement dfs on water jug problem.
   Given a 4 - liter jug filled with water & an empty 3 - liter Jug, how can one obtain exactly 2 liters in 4 liters jug. There is no measuring mark on any of them.

2. Solve the 8 puzzle Problem-using A* algorithm in Prolog.

3. Write program for Backward and forward reasoning in Prolog
4. Write a program to implement steepest ascent for 8 puzzle problem.

N. C. COLLEGE OF ENGINEERING, ISRANA

SCHEME OF STUDIES AND EXAMINATION
B. Tech. – Computer Science & Engineering
4th Year (Semester–VIII) 2015-19

Sr. No.	Course no.	Subjects		Teaching Schedule			Contact Hours	Credits
			BOS	L	T	P		
1	CSE-481	Soft Computing	CSE	3	1	-	4	4
2	CSE-48P1	Major Project	CSE	-	-	-	12	12
3	CSE-48P2	Seminar	CSE	-	-	2	2	1
4	CSE-48P3	Comprehensive Viva-Voce	CSE	-	-	-	-	2
5	CSE-482	General Proficiency and Fitness Viva	CSE	-	-	-	-	1
6	CSE-48P4	Soft Computing Lab	CSE	--	--	3	3	2
7		Elective – I	CSE	3	1	-	4	4
8		Elective – II	CSE	3	1	-	4	4
		Total		9	2	5	29	30

Elective -I

1. Fault Tolerant Computing (CSE-48E1)
2. Distributed Computing (CSE-48E2)
3. Building Enterprise Applications (CSE-48E3)
4. Agile Software Development(CSE-48E4)

Elective - II

1. Data Warehouse and Data mining (CSE-48E6)
2. Bioinformatics (CSE-48E7)
3. Embedded System (CSE-48E8)
4. Big Data & Analytics (CSE-48E9)
5. Cloud Computing (CSE-48E10)

OPEN ELECTIVE:

1. Technopreneurship (CSE-48OE1)

2. Basic of Management (MGT-481)

8th Semester (Computer Science & Engineering)

SOFT COMPUTING

CSE-481

L	T	
3	**1**	**End Semester Evaluation: 100 Marks**

On Semester Evaluation: 100 Marks

Note: - 1. There are NINE questions in a set of question-paper. All questions carry equal marks.

2. Attempt five questions in all. FIRST question is compulsory which covers the whole syllabus. Attempt ONE question from each of the other four Units.

UNIT – I

Basic concepts of neuro-computing: Artificial Neural Network (ANN) and their biological roots and motivations, Mathematical Models of Neurons, ANN architecture, Learning rules, Learning Paradigms-Supervised, Unsupervised and reinforcement Learning, ANN training Algorithms-perceptions, Training rules, Delta, Back Propagation Algorithm, Multilayer Perceptron Model, Applications of Artificial Neural Networks, Competitive learning networks, Kohonen self organizing networks, Hebbian learning; Hopfield Networks, Associative Memories, The boltzman machine; Applications.

UNIT – II

Introduction to Fuzzy Logic: Classical and Fuzzy Sets: Overview of Classical Sets, Membership Function, Fuzzy rule generation. Operations on Fuzzy Sets: Compliment, Intersections, Unions, Combinations of Operations, Aggregation Operations. Fuzzy Arithmetic: Fuzzy Numbers, Linguistic Variables, Arithmetic Operations on Intervals & Numbers, Lattice of Fuzzy Numbers, Fuzzy Equations. Fuzzy Logic: Classical Logic.

UNIT – III

Genetic algorithms(GA), Evolution strategies(Ess), Evolutionary programming(EP), Genetic Programming(GP), Selecting, crossover, mutation, schema analysis, analysis of selection algorithms; convergence; Markov & other stochastic models.

Introduction to Genetic Algorithms: Robustness of traditional optimization and search methods, The Goals of optimization, Difference between GA and traditional methods, A simple genetic algorithm, Important Similarities, Similarity Templates (Schemata), Encodings.

UNIT – IV

Random Optimization, Simulated Annealing, Tabu Search, Ant Colony Optimization, Particle Swarm Optimization, Memetic Algorithms.

Text Books:

1. S. N. Sivanandam & S. N. Deepa, Principles of Soft Computing, Wiley - India, 2007.

Reference Books:

2. Jang, Sun, Mizutani, Neuro-Fuzzy and Soft computing, Pearson.
3. Haykin, Neural networks: a comprehensive foundation, Pearson.
4. Mitchell M., An Introduction to Genetic Algorithms, Prentice-Hall, 1998.
5. Goldberg D. E., Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, 1989.
6. Klir G.J. & Yuan B., Fuzzy Sets & Fuzzy Logic, PHI.

8th Semester (Computer Science & Engineering)

FAULT TOLERANT COMPUTING

CSE-48E1

L	**T**	
3	**1**	**On Semester Evaluation: 100 Marks** **End Semester Evaluation: 100 Marks**

Note: - 1. There are NINE questions in a set of question-paper. All questions carry equal marks.

2. Attempt five questions in all. FIRST question is compulsory which covers the whole syllabus. Attempt ONE question from each of the other four Units.

UNIT-I

Fault-tolerant schemes: terminology, fundamental principles, faults and failures.

Digital circuits: Analysis and design, Redundancy, fan-out and delays, Functional and timing behavior of storage elements, Sequential circuits.

UNIT-II

Fault modeling: models at different levels, error models, system level models, Layers of Reality, Stuck-at fault model and the Single fault assumption, Functional fault models, Higher level models: Forests not trees.

Testing and test generation: basics of testing, test generation algorithms, Finding the faults, Test objective: digital/analog, logical/timing, Detection and location, Combinational test generation: The Easier Problem

UNIT-III

Concepts in fault-tolerance: hardware redundancy, information redundancy, time redundancy, and software redundancy.

Reliability/Availability modeling: reliability block diagrams, combinatorial model, MTTF, Markov model, Permanent and temporary failures, Accelerated Testing, Reliability Analysis of Redundant Systems.

UNIT-IV

Probability: What is likely to happen?, Events, disjoint, independence, Common discrete/ continuous distributions, Mean, variance, correlation, Stochastic processes: Markov, Poisson, MTBF.

System level diagnosis: Error correcting codes: Hamming codes, SED-DED codes, SEC-SBD codes, cyclic codes, Watchdog techniques, Checkpointing and error recovery., Software Fault Tolerance.

Issues in fault tolerance implementation: Voter placement, Analog/asynchronous inputs, Clock synchronization, Failed module recovery, Self-testing/duplex systems**,** Correlated failures

Recommended Books:

1. D. Siewiorek and R. Swarz, Reliable Computer Systems: Design and Evaluation, 3rd edition, AK Peters, 1998.
2. D. K. Pradhan, editor, Fault-Tolerant Computer System Design, Prentice-Hall, 1996.
3. B.W. Johnson, Design and Analysis of Fault-Tolerant Digital Systems, Addison Wesley, 1989.
4. Michael L. Bushnell and Vishwani D. Agrawal, Essentials of Electronic Testing for Digital, Memory, and Mixed-Signal VLSI Circuits, by Springer 2000.
5. Israel Koren and C. Mani Krishna, Fault-Tolerant Systems, Morgan-Kaufman Publishers, 2007.

8th Semester (Computer Science & Engineering)

DISTRIBUTED COMPUTING

CSE-48E2

L	**T**	**On Semester Evaluation: 100 Marks**
3	**1**	**End Semester Evaluation: 100 Marks**

Note: - 1. There are NINE questions in a set of question-paper. All questions carry equal marks.

2. Attempt five questions in all. FIRST question is compulsory which covers the whole syllabus. Attempt ONE question from each of the other four Units.

UNIT-I

Characterization of Distributed Systems: Introduction, Forms of computing – Monolithic, Distributed, Parallel, Cooperative, Examples of distributed systems, Resource sharing and the web, Challenges arising from the construction of distributed systems

UNIT-II

System Models: Architectural models, Client-server model, Peer-to-peer model, Variations of the above models Fundamental models, Concerned with a more formal description of the properties that are common in all of the architectural models.

UNIT-III

Interprocess Communication: The API for the internet protocols, External data representation and marshalling, Client-server communication, Group communication, Java RMI architecture-client side, server side, object registry, API for Java RMI- Remote interface, server side software, client side software, Advanced RMI- Client callback, client side, server side

UNIT-IV

Internet applications: HTML, XML, HTTP, Web form, Web session and session state data, Applets, Servlets, Web services, Simple Object Access Protocol

Textbooks And References:

1. Jean Dollimore, Tim Kindberg, George Coulouris, Distributed Systems: Concepts and Design, 4th Edition, Addison Wesley, 2005.
2. M.L. Liu, Distributed Computing: Principles and Applications, Pearson/Addison-Wesley, ISBN: 0-201-79644-9

8th Semester (Computer Science & Engineering)

BUILDING ENTERPRISE APPLICATIONS

CSE-48E3

L	T	
3	**1**	**On Semester Evaluation: 100 Marks** **End Semester Evaluation: 100 Marks**

Note: - 1. There are NINE questions in a set of question-paper. All questions carry equal marks.

2. Attempt five questions in all. FIRST question is compulsory which covers the whole syllabus. Attempt ONE question from each of the other four Units.

UNIT-I

Introduction to Enterprise application: Introduction to enterprise applications and their types, software engineering methodologies, life cycle of raising an enterprise application, introduction to skills required to build an enterprise application, key determinants of successful enterprise applications, and measuring the success of enterprise applications

Incepting enterprise application and business process modeling Inception of enterprise applications, enterprise analysis, business modeling, requirements elicitation, use case modeling, prototyping, non functional requirements, requirements validation, planning and estimation

UNIT-II

Enterprise Architecture and designing enterprise application: Concept of architecture, views and viewpoints, enterprise architecture, logical architecture, technical architecture - design, different technical layers, best practices, data architecture and design – relational, XML, and other structured data representations, Infrastructure architecture and design elements - Networking, Internetworking, and Communication Protocols, IT Hardware and Software, Middleware, Policies for Infrastructure Management, Deployment Strategy, Documentation of application architecture and design

UNIT-III

Constructing enterprise application: Construction readiness of enterprise applications - defining a construction plan, defining a package structure, setting up a configuration management plan, setting up a development environment, introduction to the concept of Software Construction Maps, construction of technical solutions layers, methodologies of code review, static code analysis, build and testing, dynamic code analysis – code profiling and code coverage

UNIT-IV

Testing and rolling out enterprise application: Types and methods of testing an enterprise application, testing levels and approaches, testing environments, integration testing, performance testing, penetration testing, usability testing, globalization testing and interface testing, user acceptance testing, rolling out an enterprise application.

Text Books:

1. Raising Enterprise Applications,, Anubhav Pradhan, Satheesha B. Nanjappa, Senthil, K. Nallasamy, Veerakumar Esakimuthu, Edition: I, Publication: Wiley India Pvt. Ltd., Year: 2010
2. Building Java Enterprise Applications, Brett McLaughlin, O'Reilly Media, first edition: 2002.

Reference Books:

1. Software Requirements: Styles & Techniques, Soren Lauesen, Software Requirements: Styles & Techniques, Addison-Wesley Professional, First edition: 2002.

2. Software Systems Requirements Engineering: In Practice, Brian Berenbach, Daniel J. Paulish, Juergen, Kazmeier, Arnold Rudorfer, Edition: I, Publication: McGraw-Hill/Osborne Media, Year: 2009.
3. Managing Software Requirements: A Use Case Approach, Dean Leffingwell, Don Widrig Edition: I, Publication: Pearson, Year: 2003,
4. Software Architecture: A Case Based Approach, Varma Vasudeva, Edition: I, Publication: Pearson, Year: 2009
5. Designing Enterprise Applications with the J2EE Platform (PDF available at http://java.sun.com/blueprints/guidelines/designing_enterprise_applications_2e/)
6. Software Testing Principles and Practices, Srinivasan Desikan, Gopalaswamy Ramesh

8th Semester (Computer Science & Engineering)

AGILE SOFTWARE DEVELOPMENT

CSE-48E4

L	T	
3	**1**	**On Semester Evaluation: 100 Marks** **End Semester Evaluation: 100 Marks**

Note: - 1. There are NINE questions in a set of question-paper. All questions carry equal marks.

2. Attempt five questions in all. FIRST question is compulsory which covers the whole syllabus. Attempt ONE question from each of the other four Units.

UNIT-I

Fundamentals of Agile:

The Genesis of Agile, Introduction and background, Agile manifesto and Principles, Extreme Programming, Feature Driven Development, Lean Software Development, Agile Project Management, Design and development practices in Agile projects, Pair Programming, Simple Design, Agile Tools, Introduction to case study on Placement Portal with agile scrum methodology.

UNIT-II

Agile Scrum Framework:

Introduction of Scrum, Project phases, Life cycle of Agile Estimation, Planning game, Product backlog, Sprint backlog, Iteration planning, User story definition, Characteristics and content of user stories, Creating product backlog & Sprint backlog of the Case study using Tool for project management, Acceptance tests and Verifying stories, Project velocity, Burn down chart, Burn up Chart, Sprint planning and retrospective, Daily scrum, Scrum roles-Product owner, Scrum Master, Scrum Team, Maintaining versions of case study with version control Tool, Case Study Integration with Continuous Integration Tool

UNIT-III

Agile Testing:

The Agile lifecycle and its impact on testing, Test- Driven Development(TDD),Development of test cases of case study, xUnit framework and tools for TDD, Testing user stories-acceptance tests and scenarios, Planning and managing testing cycle, Exploratory testing, Risk based testing, Regression tests, Test Automation, Tools to support the Agile Tester

UNIT-IV

Agile Software Design and Development & Industry Trends:

Single Responsibility Principle, Open Closed Principle, Liskov Substitution Principle, Interface Segregation Principles, Dependency Inversion Principle in Agile Design, Need and Significance of Refactoring, Refactoring Techniques, Continuous Integration, Automated build tools, Version Control, Market scenario and adoption of Agile, Agile ALM, Roles in an Agile project, Agile applicability, Agile in Distributed teams, Business benefits, Challenges in Agile, Risks and Mitigation, Agile projects on Cloud, Balancing Agility with Discipline, Agile rapid development technologies.

Text Books:

1. Agile Software Development with Scrum by Ken Schawber, Mike Beedle
2. Agile Software Development, Principles, Patterns and Practices by Robert C. Martin

Reference Books:

1. Agile Testing: A practical guide for Testers and Agile Teams by Lisa Crispin, Janet Gregory
2. Agile Software Development: The Cooperative Game by Alistair Cockburn
3. User Stories Applied for Agile Software by Mike Cohn

8th Semester (Computer Science & Engineering)

DATA WAREHOUSING AND DATA MINING

CSE-48E6

L	T	
3	**1**	**On Semester Evaluation: 100 Marks** **End Semester Evaluation: 100 Marks**

Note: - 1. There are NINE questions in a set of question-paper. All questions carry equal marks.

2. Attempt five questions in all. FIRST question is compulsory which covers the whole syllabus. Attempt ONE question from each of the other four Units.

UNIT-I

Introduction of Data Warehousing: The evolution of Data Warehousing (The Historical Context). The data warehousing – a brief history, need of data warehouse, characteristics of data warehouse, Data Marts.

Introduction of Data Mining: Motivation, importance, data mining, kind of data, functionalities, interesting patterns, classification of data mining system, major issues.

UNIT-II

Data warehouse and OLAP technology for data mining: data warehouse, operational database systems and data warehouse architecture, implementation, development of data cube technology, data warehousing to data mining, data warehouse usage.

Data Preparation: Preprocess data cleaning, data integration and transformation, data reduction, discrimination and concept hierarchy generation. Data Mining Primitives, languages and system architectures, graphical user interfaces.

UNIT-III

Concept Description: Characterization and comparison data generalization and summarization based characterization, analytical characterization, and analysis of attribute relevance, mining class comparison, and mining descriptive statistical measures in large databases.

Mining association rules in large databases, mining single dimensional Boolean association rules from transactional databases, mining multi-dimensional association rules from relational databases and data warehouses, from association to correlation analysis, constraint based association.

UNIT-IV

Application of data warehouse and data mining in government: National Data Warehouses, Other areas for data warehouse and data mining, today's development environment, looking to the future.

Case Studies: Data Warehouses in Ministry of Commerce, Data Warehouses in the World Bank.

Books and References:

1. J. Han & M. Kanber, Data Mining: Concepts and Techniques, Morgan Kaufmann/ Elsevier, India, 2001
2. D. Hand, H. Mannila, & P. Smyth. Principles of Data Mining, MIT Press, 2001.
3. M. Jarke et al. fundamentals of Data Warehouses (2nd ed.), Springer, 2003, ISBN 3-540 42089-4.

8th Semester (Computer Science & Engineering)

BIOINFORMATICS

CSE-48E7

L	**T**	
3	**1**	**On Semester Evaluation: 100 Marks** **End Semester Evaluation: 100 Marks**

Note: - 1. There are NINE questions in a set of question-paper. All questions carry equal marks.

2. Attempt five questions in all. FIRST question is compulsory which covers the whole syllabus. Attempt ONE question from each of the other four Units.

UNIT-I

Introduction to Bioinformatics:Introduction, outline of proteins, primary structure: the 20amino acids – chemical structure & properties; chirality, different types of side chain: relevanceto mutation, size , aliphatic/aromatic, polarity, charge, hydrophobicity; disulphide bonds,molecular models, polypeptide geometry: the folding chain, nomenclature, molecular graphics,Structure evolution and mutation genetic information- the triplet code; DNA structure Synthesisof proteins: cell biology background; transcription; RNA polymerase, introns, exons, splicingtranslation: ribosomes, strat/stop codons, post-translational processing

UNIT-II

Computing evolution:Phylogenetic Analysis Sequence- based taxonomy: overview andassumptions, from Multiple Alignment to phylogeny Neighbor, Joining Maximum LikelyhoodVs. Parsimony, The molecular Clock, Computer Tools for patterns, mapping and phylogeneticanalysis, Mathematical tools of proteins and nucleic acids, sequence- Function RelationshipsSequence Homology and Conserved Regions , Conserved DNA Sequences.

UNIT-III

Bioinformatics tools:Networks- WWW, CERN EMBnet; EMBL Database, SEQNET, Ge Bank, NLM ,Etc. , Sequence Databases and Sequence Analysis: Genomic , CDNA EMBLdatabase GenBank Protein sequence, Pattern recognition toolsSimilarity searching , secondary sources, genome databases, Molecular graphics software andother packages, To find sequences based on keywords & phrases, to grab individual sequences orwhole groups of Sequences from a database.

UNIT-IV

Genomics:Introduction , genome scale sequencing , comparative and evolutionary genomics,microarrays, proteomics, pharmacogenomics, Development using computer tools for sequencingprojects, PCR and restriction mapping practical and theoretical problems in sequencing. Thechallenges of whole genome sequencing, web based tools for restriction mapping, newtechnologies and new bioinformatics tools.Note: There will be 8 questions in all. Two Questions will be set from each unit. Students arerequired to attempt five questions selecting at least one question from each unit.

Books:

1. Teresa K. Attwood, David J. Parry-Smith: Introduction to Bioinformatics, 1999,Longman Higher Education, 0582327881
2. S. eddy, a. Krogh, G. Mitchison, Richard Durbin: Biological sequence analysis: probabilistic models of proteins and nucleic acids,1999, Cambridge University Press.0521629713.
3. Andreas Baxevanis, B.F. Francis Ouellete: Bioinformatics: a practical guide to

theanalysis of genes and proteins, 1998, john Wiley & sons, inc. 0471191965
4. James D. Tisdall: Beginning perl for Bioinformatics. 2001. O`reilly & Associates.0596000804
5. Michael S. Wterman: Mathematical methods for DNA sequences, 1989, CRC Press

8th Semester (Computer Science & Engineering)

EMBEDDED SYSTEMS

CSE-48E8

L	T	
3	**1**	**On Semester Evaluation: 100 Marks** **End Semester Evaluation: 100 Marks**

Note: - 1. There are NINE questions in a set of question-paper. All questions carry equal marks.

2. Attempt five questions in all. FIRST question is compulsory which covers the whole syllabus. Attempt ONE question from each of the other four Units.

UNIT-I

Introduction to an embedded systems and its design: Introduction to ES & its applications. Design parameters of an ES and its significance (with respect to all parameter), Present trends in ES, Embedded system design life cycle, product specifications and hardware, software partitioning, Co-design.

RTOS & its overview: Spell of OS 2 difference between OS 2 RTOS, Role of RTOS in ES 2 its process models (Process transition diagram), Course structure, Overview Window, CE, Unix, Mino Kernnel, UCOs & RT linux, Interrupt Roatining in RTOS & Inblow response cycle, Different IPC machines in RTOS, Scheduling construm in RTOS (hand 2 soft), Memory sowing and its protechan, Encapsulation of Semephores and Queues, Timon in RTOS (Watch dog timer)

UNIT-II

Processor Selection: Role of processor selection in ES (MP V/s Uc), Mino control-8051, 16232 bit mino controller 2 its processor, More about micro controller applications with respect to embedded system design, DSP's in ES, New trends in processing and DSP's.

Cost Compiler and cross assembly for embedded systems: Why we need cross compiler/ Assemble, Embedded software development take chain and software development tool chain, Compiler linker, locators, cross assembles, GCC compiler.

UNIT-III

Basic Concepts of Device Driving : Device drives introduction & how device are different from the normal ports, Sevical Communication enterface device drivers.

System Synthesis and Debugging Techniques:

Introduction to system synthesis & Hardware and Software, Biomulation & methods to improve to speed of simulations, Emulators (ICE) and its type, How emolutors an difference for simulations, Introduction JTAG and OCP (on chich and debugging)

UNIT-IV

Communication Protocols with reference to ES: Introduction to protocol, why we need in Es, Overview TCP(IP), UDD< wings protocols, IrDA, Blue Box, IEEE 8811

Other design issues and current trends on its application of ES

Memory optimization, Poorer optimization. Co-similation of its system on chip and SOS (System on Slices), Revision of Cost

Text and Reference Books:

1. John Catosulis, “Designing Embedded Hardware”, O’ reilly
2. An Embedded Software Primer”, David E. Simon, Pearson Education
3. Frank Vahid, Tony Givargis, “Embedded System Design” John Wiley & Sons, Inc
4. Karim Yaghmour, “Building Embedded Linux System”, O’reilly
5. Michael Barr, “Programming Embedded Systems”, O’reilly.
6. Aian C. Shaw, “Real-time system & Software”, John Wiley & sons, Inc.
7. Wayne Wolf, “Computers as Components”, Harcourt India Pvt. Ltd.

8th Semester (Computer Science & Engineering)

BIG DATA & ANALYTICS

CSE- 48E9

L	**T**	
3	**1**	**On Semester Evaluation: 100 Marks** **End Semester Evaluation: 100 Marks**

Note: - 1. There are NINE questions in a set of question-paper. All questions carry equal marks.

2. Attempt five questions in all. FIRST question is compulsory which covers the whole syllabus. Attempt ONE question from each of the other four Units.

UNIT-I

UNIT-I

Types of Digital Data: Structured, Unstructured & Semi- Structured & their issues & solutions. Introduction to OLTP and OLAP: Introduction to Knowledge Discovery, OLTP and OLAP, Different OLAP Architectures, Data models for OLAP & OLTP, Role of OLAP tools in the BI, OLAP Operations. Introduction to Big Data, Drivers for Big Data, Big Data Analytics Applications.

UNIT-II

Architecture Components: Massively Parallel Processing (MPP) Platforms, Unstructured Data Analytics and Reporting, Big Data and Single View of Customer/Product, Data Privacy Protection, Real-Time Adaptive Analytics and Decision Engines Advanced Analytics Platform, Implementation of Big Data Analytics

UNIT-III

Introduction to Hadoop, Hadoop Distributed File System, HDFS Architecture, Deamons Realated to HDFS, Working with HDFS Command ,Special Features Of Hadoop Processing Data With Hadoop, Interacting with Hadoop Ecosysten, Business Intelligence on Hadoop.

UNIT-IV

Introduction to Data Analytics tools: Pentaho Data Integration (Kettle), WEKA, Cognos.
Real World Case Studies on Big Data & Analytics.

Books:

1. Fundamentals of Business Analytics by R. N. Prasad, Seema Acharya Wiley India.
2. Introduction to Data Mining with Case Studies By GUPTA, G. K. PHI Learning
3. Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends, By Michael Minelli, Michele Chambers, Ambiga Dhiraj Wiley India.
4. Business Analytics : An application focus By RAO, PURBA HALADY, PHI Learning.
5. Big Data Analytics: Disruptive Technologies for Changing the Game by Dr. Arvind Sathi, MC Press Online.
6. Data Mining - Practical Machine Learning Tools and Techniques by Ian H. Witten, Frank Eibe, Mark A. Hall, Morgan Kaufmann Publishers.

8th Semester (Computer Science & Engineering)

CLOUD COMPUTING

CSE-48E10

L	T	
3	**1**	**On Semester Evaluation : 100 Marks** **End Semester Evaluation : 100 Marks**

Note: - 1. There are NINE questions in a set of question-paper. All questions carry equal marks.

2. Attempt five questions in all. FIRST question is compulsory which covers the whole syllabus. Attempt ONE question from each of the other four Units.

UNIT -I

Cloud Computing Overview, Applications, Intranets and the Cloud, Usage, Benefits, Limitations, Security Concerns, Regulatory Issues, Cloud Services and Providers-Google, EMC, NetApp, Microsoft, Amazon, Salesforce.com, IBM, etc. Computing Services-IaaS, PaaS, SaaS, and so on, Migration to cloud-based platform.

UNIT –II

Hardware and Infrastructure – Clients, Security, Network, Services, **Accessing the Cloud –** Platform, Web Applications, Web APIs, Web Browsers, **Cloud Storage –** Overview, Cloud Storage Providers, **Standards –** Application, Client, Infrastructure, Service

UNIT -III

Software as a Service – Overview, Driving Forces, Company Offerings, Industries, **Software plus Services –** Overview, Mobile Device Integration, Providers, **Developing Applications**

UNIT -IV

Local Clouds and Thin Clients – Virtualization, Server Solutions, Thin Clients, Case studies on latest paradigms, **Migrating to the Cloud –** Cloud Services for individuals, Force.com, Enterprise-Class Cloud Offerings, Migration, **Best Practices and the Future of Cloud Computing**

Text Books:

1. Toby Velte, Anthony Velte, and Robert Elsenpeter, "Cloud Computing, A Practical Approach", ISBN: 9780070683518, Tata Mc-Graw Hill, 2009.
2. Raj Kumar Buyya, James Broberg, and Andrezei M.Goscinski, "Cloud Computing: Principles and paradigms", 2011

Reference Books:

1. Michael Miller," Cloud Computing", 2008.
2. Joshy Joseph, and Craig Fellenstein,"Grid Computing", PearsonEducation, 2004.
3. Judith Hurwitz, Robin Bllor, Marcia Kaufman, Fern Halper, Cloud Computing for dummies, 2009.
4. Pardi, XML in Action, Web Technology,

8th Semester (Computer Science & Engineering)

CSE-48P4

SOFT COMPUTING LAB

P	**On Semester Evaluation: 120 Marks**
3	**End Semester Evaluation: 80 Marks**

1.To perform Union,Intersection and Complement operations.
2. To implement De -Morgan's Law.
3. To plot various membership functions.
4. To implement FIS Editor. Use Fuzzy toolbox to model tip value that is given after a dinner based on quality and service.
5. To implement FIS Editor.
6. Generate ANDNOT function using McCulloch - Pitts neural net.
7. Generate XOR function using McCulloch - Pitts neural net.
8. Hebb Net to classify two dimensional input patterns in bipolar with given targets.
9. Perceptron net for an AND function with bipolar inputs and targets.
10. To calculate the weights for given patterns using hetero-associative neural net.
11. To store vector in an auto-associative net. Find weight matrix & test the net with input
12. To store the vector, find the weight matrix with no self-connection. Test this using a discrete Hopfield net.