

Course No.	Course Name	L-T-P	Credits	Year of Introduction
08 CS 6012	Advanced Compiler Design	3-0-0	3	2015
<p style="text-align: center;">Course Objectives</p> <ul style="list-style-type: none"> To familiarize students of the basic structure of a typical modern compiler's back end. To give a clear understanding of the typical processes in compiler optimization 				
<p style="text-align: center;">Syllabus</p> <p>Syntax directed translation, automatic tools; Semantic analysis: symbol tables, Intermediate code generation: run-time environments, translation of language constructs; Code generation: flow graphs, register allocation, code generation algorithms; code optimization, instruction scheduling</p>				
<p style="text-align: center;">Expected Outcome</p> <p>Students who successfully complete this course will be able to</p> <ul style="list-style-type: none"> Create lexical rules and grammars for a representative programming language. Design a compiler for a concise programming language or a small subset of it. Implement semantic rules into a parser that performs attribution while parsing. 				
<p style="text-align: center;">References</p> <ol style="list-style-type: none"> <i>Compilers: Principles, Techniques and Tools</i> (2nd edition), Alfred V. Aho, Monica S. Lam, Ravi Sethi, and Jeffery D. Ullman., Addison Wesley, Boston, MA, 2006 <i>Compilers: Principles, Techniques and Tools</i>, Aho, A. V, Sethi, R. and Ullman, J. D. Pearson Education, 1986. Modern Compiler Implementation in Java, Andrew. W. Appel, Cambridge University Press, 2000. Advanced Compiler Design and Implementaiton, Steven S Muchnick, Morgan Kaufmann Publishers 				

COURSE PLAN			
Module	Contents	Hours Allotted	% of Marks in End-Semester Examination
I	Syntax Directed Translation : Syntax Directed Translation - Syntax directed definitions - Inherited and Synthesized attributes - Evaluating an SDD at the nodes of a parse tree Evaluation orders for SDDs - Dependency graphs - Ordering the evaluation of attributes - S-attributed definitions - L-attributed definitions - Semantic rules with controlled side effects - Applications of syntax directed translation Construction of syntax trees - The structure of a type - Syntax directed translation schemes - Postfix translation schemes - Parser-stack implementation of postfix SDTs - SDT's with action inside productions - Eliminating left recursion from SDTs - SDTs for L-attributed definitions. Implementing L-attributed SDDs - Translation during recursive-descent parsing - On-the-fly code generation - L-attributed SDDs and LL Parsing - Bottom up parsing of L-attributed SDDs	8	15
II	Intermediate Code Generation : Intermediate Code Generation - Intermediate Representations - Variants of syntax trees - Directed acyclic graphs for expressions - The value-number method for constructing DAGs - Three-address code - Addresses and instructions - Quadruples - Triples - Static Single-Assignment Form. Types and Declarations - Type Expressions - Type equivalence - Declarations - Storage layout for local names - Sequences of declarations - Fields in records and classes - Translation of expressions. Translation of expressions - Operations within expressions - Incremental translation - Addressing array elements - Translation of array references Control flow - Boolean expressions - Short-circuit code - flow-of-control statements - Control flow translation of Boolean expressions - Avoiding redundant Gotos - Boolean values and jumping code - Back patching - One-pass code generation using back patching - Back patching for Boolean expressions - Flow-of-control statements - Break, continue and Goto statements Translation of switch statements - syntax directed translation of switch statements - intermediate code for procedures	9	15
FIRST INTERNAL EXAM			

III	Run-Time management : Run-Time Environments - Storage organization - Static versus dynamic storage allocation - stack allocation of space - Activation trees - Activation records - Calling sequences - Variable length data on the stack Access to nonlocal data on the stack - Data access without nested procedures - Issues with nested procedures - Access links - Manipulating access links - Access links for procedure parameters - Displays Heap management - The memory manager - Locality in programs - Reducing fragmentation - Manual deallocation requests - Garbage collection - Design goals for Garbage collectors - Reference counting garbage collectors Introduction to trace-based collection - A basic mark-and-sweep collector - Basic abstraction - Optimizing mark-and-sweep - Mark-and-compact garbage collectors - Copying collectors.	9	15
IV	Code Generation : Code Generation - Issues in the design of a code generator - Instruction selection - Register allocation - Evaluation order - Target language - A simple target machine model - Program and instruction costs - Addresses in the target code - static allocation - stack allocation - run-time addresses for names Basic Blocks and Flow Graphs - Basic blocks - Next-use information - Flow graphs - Representation of flow graphs - Loops - Optimization of basic blocks - The DAG representation of basic blocks - Finding local common sub expressions - Dead code elimination - Use of algebraic identities - Representation of array references - Pointer assignments and procedure calls - Resembling basic blocks from DAGs. Simple code generator - Register and address descriptors - The code generation algorithm - Peephole optimization - Eliminating redundant loads and stores - Eliminating unreachable code - Flow-of-control optimizations - Algebraic simplification and reduction I strength - Use of machine identities Register allocation and assignment - Global register allocation - Usage counts - Register Assignment for outer loops - Register allocation for graph colouring - Instruction selection by tree rewriting - Tree-translation schemes - Code generation by tiling an input tree - Pattern matching by parsing - Routines for semantic checking - General tree matching - Optimal code generation for expressions - Ershove numbers - Generating code from Labelled Expression trees	8	15
SECOND INTERNAL EXAM			
V	Code Optimization : Code optimization - Principal sources of optimization - causes of redundancy - Semantics-preserving transformations - Global common sub expressions - Copy propagation - Dead code elimination - Code motion Induction variables and reduction in strength - Introduction to data flow analysis -The data-flow abstraction - The data-flow analysis schema - Data-flow schemas on Basic blocks - Reaching definitions Live variable analysis - Available expressions Partial redundancy elimination - The sources of redundancy - The lazy-code- motion problem - Anticipation of expressions The lazy-code-motion algorithm - Loops in Flowgraphs - Dominators - Depth- first ordering - Edges in Depth-first spanning tree - Back edges and reducibility - Depth of a flow graph - Natural loops - Convergence of Iterative data-flow algorithms Region based analysis - Regions - Region hierarchies for reducible flow graphs - Algorithm for region based analysis - Handling non-reducible flow graphs	11	20

VI	Code Scheduling : Instruction scheduling - Instruction-level parallelism - instruction pipelines and branch delays - pipelined execution - multiple instruction issue Code scheduling constraints - data dependence - dependencies among memory accesses - Trade off between register usage and parallelism Phase ordering between register allocation and code scheduling - control dependence - speculative execution support Basic-Block Scheduling - Data dependence graphs - List scheduling of basic blocks - Prioritized topological orders Global Code Scheduling - Primitive code motion - Upward code motion - Downward code motion - Updating data dependencies Global scheduling algorithms - Advanced code motion techniques - Interaction with dynamic schedulers	11	20
END SEMESTER EXAM			

Course No.	Course Name	L-T-P	Credits	Year of Introduction
08 CS 6022	Information Retrieval	3-0-0	3	2015
Course Objectives To give the Student:- <ul style="list-style-type: none"> To present the scientific underpinnings of the field of Information Search and Retrieval To learn important concepts, algorithms, and data/file structures those are necessary to design, and implement Information Retrieval (IR) systems. 				
Syllabus Introduction to information retrieval, Evaluation measures, IR models, IR query, Web Search, Link analysis, Multimedia IR, Text Categorization, Text Clustering methods				
Expected Outcome Students who successfully complete this course will understand the need and importance of information retrieval systems. Students will be able to analyze various data mining problems in terms of information retrieval and could develop various solutions based on the studied IR models and methods.				
References <ol style="list-style-type: none"> Manning .C, Raghavan P, and Schutze.H," Introduction to Information Retrieval", Cambridge University Press, 2008. Bruce Croft W, Donald Metzler, Trevor Strohman, "Search Engines: Information Retrieval in Practice" , Addison-Wesley, 2009 Ricardo Baeza-Yates and Berthier Ribeiro-Neto , "Modern Information Retrieval", Addison Wesley, 2nd edition. Karen Sparck Jones and Peter Willett," Readings in Information Retrieval" ,Morgan Kaufmann Publisher, 1997. 				

COURSE PLAN			
Module	Contents	Hours Allotted	% of Marks in End-Semester Examination
I	Information versus Data Retrieval, Basic Concepts: The Retrieval Process, Logical View of Documents. Modelling: A Taxonomy of IR Models, Performance Evaluation of IR: Recall and Precision, Other Measures, Reference Collections	6	15
II	Boolean, Vector-Space & Probabilistic Retrieval Models, Inverted Index, Index Construction, Term Weighting, Text-Similarity Metrics, TF-IDF (Term Frequency/Inverse Document Frequency) Weighting, Cosine Similarity - Vector-Space Model, Probabilistic IR Model, Binary Independence Model.	6	15
FIRST INTERNAL EXAM			
III	Query Operations and Languages, Relevance Feedback, Query Expansion, Web Search Basics, Web Crawling and Indexing, Link Analysis, Page Rank Algorithm, Hubs and Authorities.	9	15
IV	XML Retrieval, Multimedia IR: Spatial Access Methods, Distance Function, Generic Multimedia Indexing Approach.	9	15
SECOND INTERNAL EXAM			
V	Naive Bayes, Decision Trees, and Nearest Neighbor. Practical Examples.	6	20
VI	Agglomerative Clustering, K-Means, Expectation Maximization (EM), Recommender Systems.	6	20
END SEMESTER EXAM			

Course No.	Course Name	L-T-P	Credits	Year of Introduction
08 CS 6032	Evolutionary Computing	3-0-0	3	2015
<p style="text-align: center;">Course Objectives</p> <p>To give the Student:-</p> <ul style="list-style-type: none"> Knowledge of evolutionary computation techniques and methodologies set in the context of modern heuristic methods An idea of how to apply these techniques to optimisation problems and problems that require machine learning 				

Syllabus

Introduction to evolutionary computing, simulated annealing and hill climbing, genetic algorithm, ant colony optimization, particle swarm optimization, artificial bee colony optimization, case studies using various optimization problems

Expected Outcome

Students who successfully complete this course will gain knowledge in matching various evolutionary computing methods and algorithms for particular classes of computational problems.

References

1. Goldberg D E, "Genetic Algorithms in search", Optimization and machine learning, Addison-Wesley 2005.
2. Kenneth A DeJong, "Evolutionary Computation A Unified Approach", Prentice Hall of India, New Delhi, 2006.
3. Elaine Rich, Kevin Knight, "Artificial Intelligence" Tata McGraw Hill Education Private Limited, 2011.
4. Marco Dorigo and Thomas Stutzle, "Ant Colony optimization", Prentice Hall of India, New Delhi 2005.

COURSE PLAN

Module	Contents	Hours Allotted	% of Marks in End-Semester Examination
I	Historical Development, Features, Classification and Components of Evolutionary Computing, Advantages, Applications	6	15
II	Simulated Annealing: Annealing Schedule, Parameter Selection, Applications Hill Climbing: Mathematical Description, Local and Global Maxima, Ridges, Plateau, Applications	6	15
FIRST INTERNAL EXAM			
III	Genetic Algorithms: Biological Background, Schema, Theorem, GA Operators: Crossover, Mutation and Its Types-GA Algorithm, Variations Of GA: Adaptive GA and Real Coded GA	9	15

IV	Ant Colony Optimization: Ant Foraging Behavior, Theoretical Considerations, Convergence Proofs, ACO Algorithm, ACO And Model Based Search, Variations Of ACO: Elitist Ant System (EAS), Minmax Ant System (MMAS) And Rank Based Ant Colony System (RANKAS).	9	15
SECOND INTERNAL EXAM			
V	Particle Swarm Optimization: Principles of Bird Flocking and Fish Schooling , Evolution of PSO , Operating Principles , PSO Algorithm , Neighborhood Topologies , Convergence Criteria , Variations of PSO	6	20
VI	Artificial Bee Colony (ABC) Optimization: Behaviour Of Real Bees, ABC Algorithm, Variations of ABC: Abcgbest and Abcgbestdist Case Study: Travelling Salesman Problem, Knapsack Problem, N Queens	6	20
END SEMESTER EXAM			

Course No.	Course Name	L-T-P	Credits	Year of Introduction
08 CS 6042(A)	Advanced Language Technologies	3-0-0	3	2015
Course Objectives				
Gives the students an introduction about natural language Understanding and processing along with multilinguality and speech processing. To discuss about concepts like machine translation, natural language generation etc.				
Syllabus				
Natural Language Processing, Parsing, Semantics, Ambiguity, Information Retrieval, Information Extraction. Categorization, Clustering, Summarization, Multilingual Information Retrieval, Machine Translation, Dialog and Conversational Agents – Surface Realization and Discourse Planning.				
Expected Outcome				
Student must be able to understand the role of information retrieval techniques in Text and Speech processing. It also make the student to apply the various techniques and algorithms exclusively used in the area of Natural Language Processing.				

References

1. Daniel Jurafsky and James H. martin, “Speech and Language Processing”, 2000.
2. Ron Cole, J.Mariani, et al., “Survey of the State of the Art in Human Language Technology”, Cambridge University Press, 1997.

COURSE PLAN

Module	Contents	Hours Allotted	% of Marks in End-Semester Examination
I	Natural Language Processing – Linguistic Background – Classification Finite State Methods Grammar For Natural Language Processing – Parsing – Semantic and Logic Form – Ambiguity Resolution – Semantic Interpretation.	6	15
II	Information Retrieval-Information Retrieval Architecture – Indexing-Storage – Compression Techniques – Document Processing – NLP Based Information Retrieval – Information Extraction. Categorization – Extraction Based Categorization	6	15
FIRST INTERNAL EXAM			
III	Clustering- Hierarchical Clustering- Document Classification and Routing- Finding and Organizing Answers From Text Search – Use Of Categories and Clusters For Organizing Retrieval Results – Text Categorization and Efficient Summarization Using Lexical Chains – Pattern Extraction.	9	15
IV	Multilinguality – Multilingual Information Retrieval and Speech Processing – Multimodality – Text and Images – Modality Integration.	9	15
SECOND INTERNAL EXAM			
V	Machine Translation – Transfer Metaphor – Interlingual and Statistical Approaches – Discourse Processing	6	20
VI	Dialog and Conversational Agents Natural Language Generation – Surface Realization and Discourse Planning.	6	20
END SEMESTER EXAM			

Course No.	Course Name	L-T-P	Credits	Year of Introduction
08 CS 6042(B)	Big Data Essentials	3-0-0	3	2015
<p style="text-align: center;">Course Objectives</p> <p>To give the Student:- An understanding of problems and solutions related with Big Data domain. To familiarize with the latest data intensive computing systems which can handle the big data challenge.</p>				
<p style="text-align: center;">Syllabus</p> <p>Importance and features of Big Data, Data intensive computing using Hadoop and MapReduce, Distributed data storage systems, Data mining with big data ,recent advancements in big data analytics.</p>				
<p style="text-align: center;">Expected Outcome</p> <p>Students who successfully complete this course will understand the challenges faced by both scientific and business community due to arise of Big Data. Students will be able to analyze a data intensive problem and could select a suitable big data technology to find efficient solutions to such problems.</p>				
<p style="text-align: center;">References</p> <ol style="list-style-type: none"> 1. Runkler, Thomas A, "Data Analytics - Models and Algorithms for Intelligent Data Analysis", Springer, 2012. 2. Nathan Marz , "Big Data: Principles and best practices of scalable real-time data systems",Manning Publications, 2012Noreen Burlingame ,"Little Book of Big Data" Ed. 2012 3. Tom White " Hadoop: The Definitive Guide" Third Edition, O'reilly Media, 2012. 4. Jimmy Lin, Chris Dyer "<u>Data-Intensive Text Processing with MapReduce</u>" Morgan & Claypool Publishers, 2010. 5. R. Owens, Brian Femiano, and Jon Lentz "Hadoop Real World Solutions Cookbook Jonathan" 6. Anand Rajaraman and Jeffrey David Ullman, "Mining of Massive Datasets", Cambridge University Press, 2012. 7. <u>Xindong Wu</u> ,<u>Xingquan Zhu</u> ,<u>Gong-Qing Wu</u> ,<u>Wei Ding</u> "Data Mining with Big Data" IEEE Transaction of Knowledge and Data Engineering Vol. 26, No. 1, January 2014 pp.97-107 8. Tomasz, K. , Przemyslaw, K. and Wojciech, I.(2014), ' Parallel processing of large graphs' . Future Generation Computer Systems,Elsevier Vol. 32, pp. 324–337. 9. Malewicz, G.,Austern, M.H., Bik, A.J.C., Dehnert, J.C., Horn, I., Leiser, N. and CzajkowskiG. (2010), Pregel: a system for large-scale graph processing'. Proceedings of the 2010 International Conference on Management of Data, SIGMOD '10, New York, USA, pp. 135-146 10. www.serendio.com/resources/smart-meter-analytics-platform-selection.pdf 				

COURSE PLAN			
Module	Contents	Hours Allotted	% of Marks in End-Semester Examination
I	Big Data: Characteristics, Importance, Use cases: Fraud detection patterns, Data Analytics Lifecycle, Role of Data Scientist, Big Data Sources.	6	15
II	Hadoop & Mapreduce- Hadoop architecture, Hadoop distributed file system, MapReduce framework, Hadoop Input/output formats, Setting up a Hadoop cluster.	7	15
FIRST INTERNAL EXAM			
III	MapReduce Programming: Mapreduce Algorithm Design, Data Flow In Map Reduce Pipeline, Partitioners and Combiners, Developing A Mapreduce Application.	8	15
IV	Data Storage: NoSQL Stores , Key-Value Stores , Columnar Stores , Document Stores ,Graph Databases ,Case Studies , HDFS, HBase, Hive, MongoDB, Neo4j	7	15
SECOND INTERNAL EXAM			
V	Big Data Mining: Data Mining with Big Data, Issues, Case Study, Clustering on Big Data, Limitations of Mapreduce Framework, Case Study-Graph Algorithms on Mapreduce.	7	20
VI	Advanced Big Data Analytics: Real time data analytics, Alternate models for big data analytics, In memory models, Bulk Synchronous Parallel-case study-apache Giraph.	7	20
END SEMESTER EXAM			

Course No.	Course Name	L-T-P	Credits	Year of Introduction
08 CS 6042(C)	Algorithms and Complexity	3-0-0	3	2015
<p style="text-align: center;">Course Objectives</p> <p>To provide an introduction to the different complex algorithms in computer programming such as graph algorithms, randomized algorithms etc and the complexity classes such as NP-Hard and NP- Complete problems.</p>				

Syllabus

Analysis of RAM model, Recurrence analysis, Advanced Data Structures, Graph Algorithms and complexity, Randomized Algorithms, Complexity classes, Approximation algorithms, Probabilistic Complexity Classes, Probabilistic Proof Theory and Certificates.

Expected Outcome

- Students will understand different complex algorithms in computer programming.
- Students will be able to identify the complexity classes.

References

1. Dexter Kozen, The Design and Analysis of Algorithms, Springer, 1992.
2. T. H. Cormen, C. E. Leiserson, R. L. Rivest, Introduction to Algorithms, Prentice Hall India, 1990.
3. S. Basse, Computer Algorithms: Introduction to Design and Analysis, Addison Wesley, 1998.
4. U. Manber, Introduction to Algorithms: A creative approach, Addison Wesley, 1989.
5. V. Aho, J. E. Hopcraft, J. D. Ullman, The design and Analysis of Computer Algorithms, Addison Wesley, 1974.
6. R. Motwani and P. Raghavan, Randomized Algorithms, Cambridge University Press, 1995.
7. C. H. Papadimitriou, Computational Complexity, Addison Wesley, 1994
8. Leonard Adleman. Two theorems on random polynomial time. In Proceedings of the 19th IEEE Symposium on Foundations of Computer Science, pages 75–83, 1978.
9. J. Gill. Computational complexity of probabilistic Turing machines. SIAM Journal of Computing, 6:675–695, 1977.
10. C. Lautemann. BPP and the Polynomial Hierarchy. Information Processing Letters, 17:215–217, 1983.
11. M. Sipser. A complexity theoretic approach to randomness. In Proceedings of the 15th ACM Symposium on Theory of Computing, pages 330–335, 1983.
12. L.G. Valiant and V.V. Vazirani. NP is as easy as detecting unique solutions. Theoretical Computer Science, 47:85–93, 1986.

COURSE PLAN

Module	Contents	Hours Allotted	% of Marks in End-Semester Examination
--------	----------	----------------	--

Kerala Technological University
Master of Technology – Curriculum, Syllabus & Course Plan

I	Analysis: RAM model - Notations, Recurrence analysis - Master's theorem and its proof - Amortized analysis	6	15
II	Advanced Data Structures: B-Trees, Binomial Heaps, Fibonacci Heaps, Disjoint Sets, Union by Rank and Path Compression	7	15
FIRST INTERNAL EXAM			
III	Graph Algorithms and complexity: Matroid Theory, All-Pairs Shortest Paths, Maximum Flow and Bipartite Matching.	6	15
IV	Randomized Algorithms : Finger Printing, Pattern Matching, Graph Problems, Algebraic Methods, Probabilistic Primality Testing, De-Randomization	8	15
SECOND INTERNAL EXAM			
V	Complexity classes - NP-Hard and NP-complete Problems -Cook's theorem NP completeness reductions. Approximation algorithms - Polynomial Time and Fully Polynomial time Approximation Schemes.	8	20
VI	Probabilistic Complexity Classes, Probabilistic Proof Theory and Certificates.	7	20
END SEMESTER EXAM			

Course No.	Course Name	L-T-P	Credits	Year of Introduction
08 CS 6042(D)	Software Architecture and Design	3-0-0	3	2015
Course Objectives				
<p>To give the Student:-</p> <ul style="list-style-type: none"> In-depth knowledge in basic architecture and design so as to enable the students to understand the subjects. 				
Syllabus				
Introduction; Middleware Architectures; Software Architectural Process and Documentation; Aspect Oriented Architecture and MDA; Web Services; Architectural Patterns.				
Expected Outcome				
Students who successfully complete this course will have the ability to design and experiment with software prototypes; Students will be able to create and document the overall design for a system and document this design using UML and other methodologies and notations.				

References

1. Ian Gorton, "Essential Software Architecture", Springer, 2011.
2. Frank Buschmann, Regine Muiner, Hans Rohnert, Peter Sommerlad, Michael Stal, "Pattern Oriented Software Architecture", Volume 1, John Wiley, 2001
3. George Fairbanks, "Just enough Software Architecture", Marshall and Bainerd, 2010.
4. Simon Brown, "Software Architecture for Developers", Lean Publishing, 2013.
5. Mahesh P Matha "Object Oriented Analysis & Design Using Uml An Introduction To Unified Process & Design Patterns", Prentice-hall Of India Pvt Ltd, 2010

COURSE PLAN			
Module	Contents	Hours Allotted	% of Marks in End-Semester Examination
I	Introduction Definition – architecture for non-functional requirements – role of Software architect – Technologies – Software Quality attributes	9	15
II	MIDDLEWARE ARCHITECTURES - Classification - distributed objects - Message oriented middleware - application servers - Enterprise java Beans Architecture. Service Oriented Architecture-Service Oriented Systems .	9	15
FIRST INTERNAL EXAM			
III	SOFTWARE ARCHITECTURAL PROCESS AND DOCUMENTATION: Process – Requirements, design, validation-Documentation - UML 2.0, Architectural views, component diagrams, templates.	9	15
IV	ASPECT ORIENTED ARCHITECTURE AND MDA: Aspect Oriented Architecture-Aspects, AOP, example- architecture, Aspects and middleware, Tools, Model Driven Architecture-need, tools, MDA and Software Architecture- requirements, transformation	9	15
SECOND INTERNAL EXAM			
V	Web Services – components – Restful Web Services- Advanced middleware architectures –Business Process Orchestration, Integrating Architecture issues, ESB – Message brokers	10	20

VI	Architectural Patterns -Patterns in Software Architecture – layers, pipes and filters, blackboard, broker, MVC, Presentation-abstraction-control, Adaptable systems	10	20
END SEMESTER EXAM			

Course No.	Course Name	L-T-P	Credits	Year of Introduction
08 CS 6052(A)	Cloud Computing	3-0-0	3	2015
Course Objectives				
<ul style="list-style-type: none"> • To understand the concept of cloud and utility computing • To understand the various issues in cloud computing • To familiarise themselves with the lead players in cloud • To appreciate the emergence of cloud as the next generation computing paradigm 				
Syllabus				
Evolution of cloud computing and its architecture, basics of virtualization and its types, cloud infrastucture, programming model, security in cloud				
Expected Outcome				
At the end of this course the student will be able to appreciate the new computing model called cloud computing and why it is creating such a hype in the 21 st century. He/She will be able to use the open source cloud services. Understand that one of the major issues in usage of public cloud is security. The student will have the knowledge to deploy a private cloud and understand the issues currently prevailing.				

References

1. Distributed and Cloud Computing, From Parallel Processing to the Internet of by Kai Hwang, Geoffrey C Fox, Jack G Dongarra, Morgan Kaufmann Publishers, 2012.
2. Cloud Computing: Implementation, Management, and Security by John W.Rittinghouse and James F.Ransome : CRC Press 2010
3. Cloud Computing, A Practical Approach by Toby Velte, Anthony Velte, Robert Elsenpeter: TMH, 2009
4. Cloud Application Architectures: Building Applications and Infrastructure in the Cloud: Transactional Systems for EC2 and Beyond (Theory in Practice (O'Reilly)) by George Reese: O'Reilly
5. James E. Smith, Ravi Nair, Virtual Machines: Versatile Platforms for Systems and Processes, Elsevier/Morgan Kaufmann, 2005.
6. Katarina Stanoevska-Slabeva, Thomas Wozniak, Santi Ristol, "Grid and Cloud Computing – A Business Perspective on Technology and Applications", Springer.

COURSE PLAN

Module	Contents	Hours Allotted	% of Marks in End-Semester Examination
I	Evolution of Cloud Computing Evolution of Cloud Computing -System Models for Distributed and Cloud Computing - NIST Cloud Computing Reference Architecture -IaaS - On-demand provisioning - Elasticity in cloud - Egs of IaaS providers - PaaS - Egs. Of PaaS providers - SaaS - Egs. Of SaaS providers - Public, Private and Hybrid clouds.)	6	15
II	Virtualization: Basics of virtualization - Types of Virtualization - Implementation Levels of Virtualization - Virtualization Structures/ Tools and Mechanisms - Virtualization of CPU, Memory, I/O Devices - Desktop virtualization - Server Virtualization.	9	15
FIRST INTERNAL EXAM			
III	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.	9	15

IV	Programming Model: Parallel and Distributed programming Paradigms – MapReduce , Twister and Iterative MapReduce – Hadoop Library from Apache – Mapping Applications	6	15
SECOND INTERNAL EXAM			
V	Programming Support - Google App Engine, Amazon AWS - Cloud Software Environments - Eucalyptus, Open nebula, OpenStack.	6	20
VI	Security in Cloud: Security Overview – Cloud Security Challenges – Software-as-a-Service Security – Security Governance – Risk Management – Security Monitoring – Security Architecture Design – Data Security – Application Security – Virtual Machine Security	6	20
END SEMESTER EXAM			

Course No.	Course Name	L-T-P	Credits	Year of Introduction
08 CS 6052(B)	Data Compression	2-1-0	3	2015
Course Objectives				
To familiarize the students with the different data compression techniques for image compression, audio compression, video compression etc. It helps to analyze different compression algorithms and their implementation.				
Syllabus				
Overview of basic techniques, compression algorithms, transformation methods, video compression, audio compression and their implementation				
Expected Outcome				
Students are able to implement various compression techniques on text, images and video. They can evaluate the performance comparison of different compression algorithms based on number of inputs given .				

References

1. . David Solomon, Data compression: the complete reference, 2nd edition, Springer-Verlag, New York. 2000.
2. Stephen Welstead, Fractal and wavelet Image Compression techniques, PHI, New Delhi-1, 1999.
3. Khalid Sayood, Introduction to data compression, Morgan Kaufmann Publishers, 2003 reprint.

COURSE PLAN

Module	Contents	Hours Allotted	% of Marks in End-Semester Examination
I	Introduction, Basic Techniques :- Intuitive Compression - RLE - RLE Text Compression - RLE Image Compression - Move-to-Front Coding, Statistical Methods: - The Golomb Code - Shannon-Fano Coding - Huffman Coding - Adaptive Huffman Coding.	6	15
II	Dictionary Methods: - String Compression - Simple Dictionary Compression - LZ77 - LZSS - LZ78 - LZW - LZWM - LZAP - LZP - RAR and WinRAR - Deflate: Zip and Gzip.	9	15
FIRST INTERNAL EXAM			
III	Transform based techniques: Image Transforms - Orthogonal Transforms - DCT Transform - JPEG - JPEG-LS, Wavelet Methods- Fourier transform - Fourier Image Compression - CWT and its Inverse - The Haar Transform - The DWT.	9	15
IV	Video compression: Analog Video - Composite and Components Video - Digital Video - Video Compression - MPEG - MPEG-4 - H.261	6	15
SECOND INTERNAL EXAM			
V	Audio Compression: Sound - Digital Audio - The Human Auditory System - μ -Law and A-Law Companding - ADPCM Audio Compression - MLP Audio - Speech Compression - Shorten - MPEG-4 Audio Lossless Coding.	6	20
VI	Comparison of compression algorithms, Implementation of compression algorithms.	6	20
END SEMESTER EXAM			

Course No.	Course Name	L-T-P	Credits	Year of Introduction
08 CS 6052(C)	Bio-Informatics	3-0-0	3	2015
Course Objectives				
To provide an overview of algorithms used in Bioinformatics.				
Syllabus				
Algorithms and Complexity, Molecular Biology Primer, Exhaustive Search, Greedy Algorithms, Dynamic Programming Algorithms, Divide-and-Conquer Algorithms, Graph Algorithms, Combinatorial PatternMatching, Clustering and Trees, HiddenMarkovModels, Randomized Algorithms.				
Expected Outcome				
At the end of the course students will be able to				
References				
1. Neil C. Jones and Pavel A. Pevzner, An Introduction to BioinformaticsAlgorithms, MIT Press, First Indian Reprint 2005. 2. Gary Benson Roderic page (Eds), Algorithms in Bioinformatics, Springer International Edition, First Indian Reprint 2004. 3. Gusfields G, Algorithms on strings, trees and sequences- Computer Science and Computational Biology, Cambridge University Press 1997. 4. T. K. Attwood and D. J. Parry-Smith, Introduction to Bioinformatics, Pearson Education, 2003.				
COURSE PLAN				
Module	Contents	Hours Allotted	% of Marks in End-Semester Examination	
I	Algorithms and Complexity- Biological algorithms versus computer algorithms – The change problem –Correct versus Incorrect Algorithms – Recursive Algorithms – Iterative versus Recursive Algorithms – Big-O Notations – Algorithm Design Techniques.	8	15	
II	Molecular Biology Primer –Structure of DNA–Proteins - Analyzing DNA – Species - Exhaustive Search – Mapping Algorithms – Motif-Search Trees – Finding Motifs – Finding a Median String .	7	15	
FIRST INTERNAL EXAM				

Kerala Technological University
Master of Technology – Curriculum, Syllabus & Course Plan

III	Greedy Algorithms- Genome Rearrangements – Sorting by Reversals – Approximation Algorithms – A Greedy Approach to Motif Finding – DNA Sequence comparison – Manhattan Tourist Problem – Edit Distance and Alignments – Longest Commons Subsequences- Global Sequence Alignment – Scoring Alignment – Local Sequence Alignment – Alignment with Gap Penalties – Multiple Alignment-Gene Predictions – Approaches to Gene Prediction - Spiced Alignment	8	15
IV	Divide and Conquer Algorithms - Graphs – Graphs and Genetics – DNA Sequencing – Shortest Superstring Problem – DNA arrays as an alternative sequencing techniques – Sequencing by Hybridization – Path Problems – Fragment assembly in DNA Sequencing – Protein Sequencing and Identification – The Peptide Sequencing Problem – Spectrum Graphs – Spectral Convolution and Alignment – Combinatorial Pattern matching..	7	15
SECOND INTERNAL EXAM			
V	Clustering and trees – Gene expression analysis – Hierarchical clustering- k-means clustering – Clustering and corrupted Cliques – Evolutionary Trees – Distance-based tree reconstruction – Reconstruction trees from additive matrices – Evolutionary trees and hierarchical clustering – Character-based tree reconstruction – Small and large Parsimony Problem.	7	20
VI	Hidden Markov Models-CG-Islands and the “Fair Bet Casino” - The Fair Bet Casino and Hidden Markov Models - Decoding Algorithm - HMM Parameter Estimation - Profile HMM Alignment - Randomized Algorithms - Gibbs Sampling - Random Projections.	6	20
END SEMESTER EXAM			

Course No.	Course Name	L-T-P	Credits	Year of Introduction
08CS6062(P)	Mini Project	0-0-4	2	2015

Course Objectives

To make students design and develop a system or application in the area of their specialization.

Approach

In this practical course, each student is expected to design and develop a moderately complex computer / information system with practical/research applications; Students can take up any application level/system level project pertaining to a relevant domain, preferably based on papers from IEEE/ACM journals. Projects can be chosen either from the list provided by the faculty or in the field of interest of the student. A committee consisting of minimum three faculty members specialised in computer science and engineering will perform assessment of the mini project.. Marks will be awarded based on the report and their performance during presentations and demonstrations. Publishing the work in Conference Proceedings/Journals with National/International status with the consent of the guide will carry an additional weightage in the evaluation process.

Expected Outcome

Upon successful completion of the miniproject, the student should be able to

1. Identify and solve various problems associated with designing and implementing a system or application.
2. Test the designed system or application.

Course No.	Course Name	L-T-P	Credits	Year of Introduction
08CS6072(P)	Data Mining and Analytics Lab	0-0-2	1	2015

Syllabus

Experiments are based on topics covered in Data Mining

To give the Student:-

- The ability to design and implement various algorithms related to information retrieval.
- Hands on experience with latest data storage and analytic infrastructure like Hadoop and R

08CS6072(P) - Experiments

Experiment No	Description
I	Study of WEKA / R Tool
II	Building an inverted index
III	Implementation of text classification techniques
IV	Implementation of text clustering techniques
V	Implementation of Page Rank Algorithm
VI	Setting up a Hadoop Cluster
VII	Developing a MapReduce Application
VIII	Working with a distributed data storage(Any NoSQL model)