Web Engineering
COT 402

L T P Theory: 100
4 1 - Sessional: 50

1. Information Architecture

2. Dynamic HTML and Web Designing

3. CGI using PERL

4. Java Server Pages
Basics, Integrating Scripts in JSPs, JSP Objects and Components, configuring and troubleshooting, JSP: Request and response objects, Retrieving the contents of an HTML form, Retrieving a Query String, Working with Beans, Cookies, Creating and Reading Cookies. Using Application Objects and Events.

5. XML
Relationship between HTML, SGML and XML, Basic XML, Valid Documents, Ways to use XML, XML for Data Files, Embedding XML into HTML documents, Converting XML to HTML for DISPLAY, Displaying XML using CSS and XSL, Rewriting HTML as XML, The future of XML.

Note:- At least one question will be set from each unit.
2. Scott Guelich, Shishir Gundavaram, Gunther Birzniek; CGI Programming with Perl 2/e, O'Reilly
3. Doug Tidwell, James Snell, Pavel Kulchenko; Programming Web Services with SOAP, O'Reilly
5. Yong, XML Step by Step, PHI.
Compiler Design
COT-404

L  T  P  Theory:  100
4  1  -  Sessionals:  25

1. Introduction to System Software
   Introduction and types of assemblers, linkers, loaders, compilers and translators, the structure of a compiler, different states in the construction of a compiler.

2. Lexical Analysis
   Design of lexical analyzer.

3(i). Basic Parsing Techniques
   Parsers, shift-reduce parsing, operator- precedence parsing, top-down parsing, predictive parsers.

3(ii). Automatic Construction of Efficient Parsers
   L.R. Parsers, the canonical collection of L R (O) items, construction of SLR parsing tables, constructing canonical L.R. Parsing tables, Constructing LALR parsing tables, implementation of L R Parsing tables.

4(i). 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.

4(ii). Symbol Tables
   The contents of a symbol table, data structures for symbol tables, representing scope information.

5(i). Run Time Storage Administration
   Implementation of a simple stack allocation scheme, implementation of block structured languages, storage allocation in FORTRAN storage allocation in block structured language.

5(ii). Error Detection And Recovery
   Error, Lexical-phase errors, syntactic- phase errors, semantic errors.

6(i). Introduction To Code Optimization
   The principle sources of optimization, loop optimization, the DAG representation of basic blocks, value number and algebraic laws, global data-flow analysis,

6(ii). Code Generation
   Object programs, problems in code generation, a machine model, a simple code generator, register allocation and assignment, code generation from DAGs, peephole optimization.

Note:- At least one question will be sent from each unit.

BOOKS
2. Donovan, J, System Programming, TMH.
1. Chalk out the storyboard and design of Dairy Food Limited. As the name reflects your site provides dairy products and aims at opening an online store. Your storyboard should cover all the features that you plan to have on the site.
2. Create your own page with your favourite hobbies.
3. Create a Menu or a Table of content web page. Each menu item or section of the table of content should load a different web page. For example, if the user clicks on Menu one or section 1 then the link should take him to respective menu html. Or section and so on.
4. Create a web site for your College,
5. Create a frameset that is divided into three sections. The frameset should have three zones.

- the topmost section of the frameset should take up about just 15% of the browser window. Name this frame title.
- The middle section should be 70% of the browser window. Name this frame title.
- The lower most sections should also be about 15% of the browser window. Name this section as menu. Create pages for each section. For the lowermost section, create page that loads the content into the middle section. The topmost section should contain a page describing the web page itself.
6. Create a web page, which displays the map of your Country Link, each city/state on the image using image map, such that the respective HTML page of the city/state is displayed when the user selects an area.
7. Add the tickertape applet to your page by customizing it for the following settings:
   - Increase the count by one.
   - Accordingly update the message count.
   - Change the text color to (237,192,171)
   - Experiment with changing the scrolling speed.
   - Customize the message text as per your page requirement.
8. Incorporate a quest book into the Dairy Food Webpage and use Java Script to build validations into the form.
9. Use Stylesheet to modify the following:
   - Change background to modify the following.
   - Change font, type, face and color.
   - Align Text.
   - Remove underlines from hyperlinks.
10. Use Microsoft’s Personal Web Server to set up your website.
Distributed Operating Systems
COT-440
(Departmental Elective III)

L T P Theory: 75
3 1 - Sessional: 50

1. Architecture of distributed O.S.
Introduction, motivation, system architecture type, issues in distributed O.S., Communication primitive.

2. Distributed mutual Inclusion

3. Distributed dead lock detection
Introduction, dead lock handling strategies, issues in deadlock detection & resolution, Control organization, centralized, distributed & hierarchical detection algorithm.

4. Distributed file system
Introduction, architecture mechanism for building, design issues, log structured file system.

5. Distributed Scheduling
Introduction, motivation, issues in load distribution, component of load algorithm, stabilizing load distribution algorithm, performance comparison, selection of a suitable load sharing algorithm, requirement for load distribution, task migration, issues in task migration.

Note:- At least one question will be set from each unit

BOOKS
2. A S Tanenbann: Modern operating systems, PHI.
Software Project Measurement  
COT-442  
(Departmental Elective III)

<table>
<thead>
<tr>
<th>L</th>
<th>T</th>
<th>P</th>
<th>Theory:</th>
<th>Sessional:</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>1</td>
<td>-</td>
<td>75</td>
<td>50</td>
</tr>
</tbody>
</table>

1. **Software Project Planning**  
Cost estimation models single use multi variable; COCOMO model; COCOMO II model; Putnam model; Delphi model; S/W configuration management; risk management.

2. **Internal and External Attribute Management**  
Size, length, reuse, functionality, complexity, types of structural measures, control flow structures, modularity and information flow attributes, object oriented metrics, data structure, measuring external product attributes- modeling, S/W quality, measuring aspects of quality.

3. **Object-Oriented Design Measurement**  
Cohesion & Coupling Consideration; Coupling – interaction, Component, inheritance; different types of each of these couplings & their measurements; Cohesion- method, class, inheritances; types of each of these Cohesions & their measurements.

**Note:** Atleast two questions will be set from each unit.

**BOOKS**  
1. **Introduction**
   Origin, imposition, representation, role of knowledge, use of prolog for Natural Language Processing (NLP), Finite State Transition Networks(FSTN), notation, representation and traversal of FSTN in Prolog, Finite State Transducers(FST), implementation in Prolog, limitation of SM.

2. **Recursive and Augmented Transition Networks (RTN)**
   Modeling recursion, representation, traversal, implementation in Prolog, push down transducers, implementation, advantage and limitations of RTN, augmented transition networks.

3. **Grammar and Parsing**
   Grammar as knowledge representation, words, rules, structures, representation in Prolog, subcategorization, definite clause grammars, classes of grammars and languages, top down and bottom up parsing, comparison strategies, BFS and DFS, storing intermediate results, ambiguity, determinism and lookahead.

4. **Well formed Sub-string tables and Charts**
   Well formed substring tables, active charts, rules of chart parsing, initialization, rule invocation, house keeping, implementation of top down and bottom up chart parsers, search strategy, alternative rule invocation, implementing flexible control, efficiency.

5. **Features and the Lexicon**
   Feature theoretic syntax, feature structures as graphs, feature structures in Prolog, subsumption and unification, the status of rules, implementing PATR in Prolog, chart parsing with feature-based grammars, representation of lexical knowledge, implementing a lexicon in Prolog, DAGs versus terms.

6. **Semantics**
   Compositionality, meaning as reference, translation to a meaning representation language, computational semantics as feature instantiation, transitive verbs and quantification, ambiguity, preferences and timing, building semantic checking in to the grammar.

7. **Question answering and Inference**
   Question answering, evaluating DBQ formulae, standard logical inference, implementing forwards inference in Prolog, the pathological nature of logical inference, primitives and canonical forms, classes and inheritance, plausible inference and defaults.

**Note:** At least one question will be set from each unit.

**BOOKS**
1. Gerald Gazdar and Chris Mellish: Natural Language Processing in Prolog, Addison Wesley
2. Allen James: Natural Language Understanding, Benjamin Cummins
4. Schwartz, Steven C.: Applied Natural Language Processing, Petrocelli
Bioinformatics
COT-446
(Departmental Elective III)

L  T  P        Theory:  75
3  1 -          Sessional:  50

1. Introduction to Bioinformatics:
Introduction, outline of proteins, primary structure: the 20 amino acids - chemical structure & properties: chirality, different types of side chain: relevance to 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 Synthesis of proteins: cell biology background; transcription: RNA polymerase, introns, exons, splicing translation: ribosomes, start/stop codons, Post-translational processing

2. Computing Evolution:
Mathematical tools of proteins and nucleic acids, sequence-Function Relationships Sequence Homology and Conserved Regions, Conserved DNA sequences.

3. Bioinformatics tools:
Networks - WWW, CERN EMBnet; EMBL Database, SEQNET, GenBank, NLM, etc., Sequence Databases and Sequence Analysis: genomic, cDNA EMBL database GenBank protein sequence, pattern recognition tools
Similarity searching, secondary sources, Genome databases, Molecular graphics software and other packages, To find sequences based on keywords & phrases, to grab individual sequences or whole groups of sequences from a database

4. Genomics:
Introduction, genome scale sequencing, comparative and evolutionary genomics, microarrays, proteomics, pharmacogenomics, Development using computer tools for sequencing projects, PCR and restriction mapping, practical and theoretical problems in sequencing, The challenges of whole genome sequencing, Web-based tools for restriction mapping, new technologies and new bioinformatics tools.

BOOKS
1. Introductory Concepts
Operating system functions and characteristics, historical evolution of operating systems, Real
time systems, Distributed systems, Methodologies for implementation of O/S service system
calls, system programs, Interrupt mechanisms.

2. File Systems
Functions of the system, File access and allocation methods, Director Systems: Structured
Organization, directory and file protection mechanisms, implementation issues; hierarchy of file
and device management.

3. CPU Scheduling
Levels of Scheduling, Comparative study of scheduling algorithms, multiple processor
scheduling.

4. Storage Management
Storage allocation methods: Single contiguous allocation, Multiple contiguous allocation,
Paging, Segmentation combination of Paging and Segmentation, Virtual memory concepts,
Demand Paging, Page replacement Algorithms, Thrashing,

5. Device Management
Hardware Organization, Device scheduling policies

6. Deadlocks
Deadlock characterization, Deadlock prevention and avoidance, Deadlock detection and
recovery, practical considerations.

7. Concurrent Processes
Critical section problem, Semaphores, Classical process coordination problems and their
solutions, Interprocess Communications.

8. Protection
Mechanisms and Policies, Implementation.

Note:- The question paper will contain 8 questions in all. The student will be required to answer
any five. At the most one question will be set from section 1,5,8, and atleast one question each
will be set from rest of units

BOOKS
Fundamentals of Computer Networks
COT-474
(Open Elective II – For branches except CO, IT, EC)

<table>
<thead>
<tr>
<th>L</th>
<th>T</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>1</td>
<td>-</td>
</tr>
</tbody>
</table>

Theory: 75  
Sessional: 50

1. **Introduction**  
Network Functions, Network Topology, Network Services, Switching Approaches, Transmission media and systems, multiplexing and signaling techniques, Error detection and correction, ISDN and BISDN

2. **Layered Architectures**  
Examples, OSI Reference Model, Overview of TCP/IP architecture, Socket system calls, SNMP, Electronic Mail.

3. **Peer-to-Peer Protocols**  
Protocols, Service Models and End-to-End requirements, ARQ, Sliding Window, RTP, HDLC, PPP protocols, Statistical Multiplexing.

4. **MAC and LAN Protocols**  
Multiple access communication, Random Access-ALOHA, Slotted-ALOHA, CSMA, CSMA-CD, Channelization – FDMA, TDMA, CDMA, Channelization in Cellular networks LAN Standards - 802.3, 802.4, 802.5, 802.6, FDDI, 802.11, LAN Bridges.

5. **Packet Switching Networks**  

6. **TCP/IP**  

**Note:**- There will be at least one question from each unit.

**BOOKS**
4. William Stallings: Data and Computer Communications 5/e, PHI.
Object Oriented Software Engineering
(Open Elective II - For all branches)
COT-476

L    T    P    Theory:  75
3    1    -    Sessional:  50

1. Review of Object Oriented Systems
Design Objects, Class hierarchy, inheritance, polymorphism, object relationships and
associations, aggregations and object containment, object persistence, meta-classes, Object-
oriented systems development life cycle, Software development process Object Oriented systems
development: a use-case driven approach.

2. Methodology for Object Oriented Design
Object modeling technique as software engineering methodology, Rumbaugh methodology,
Jacobson Methodology, Booch Methodology, Patterns, Frameworks, the unified approach,
unified modeling language (UML).

3. Object Oriented Analysis
Analysis Process, Use-Case Driven Object Oriented Analysis, Use-Case Model, Object
Classification, Theory, Different Approaches for identifying classes, Classes, Responsibilities
and Collaborators, Identifying Object Relationships, Attributes and Methods, Super-sub Class
Relationships, A-Part of Relationships-Aggregation, Class Responsibilities, Object
Responsibilities.

4. Object Oriented Design
Object oriented design process, corollaries, design axioms, design patterns, object oriented
design philosophy, UML Object Constraint Language, Designing Classes: The Process, Class
Visibility, Refining Attributes, Designing Methods and Protocols, Packages and Managing
classes,
Designing Interface Objects, View layer interface design, Macro and Micro level interface
design process.

Note:- Two questions will be set from each unit.

BOOKS
Ivar Jacobson, Object Oriented Software Engineering, Addison-Wesley, 1995.
1. **Introduction to Expert System**
   Features of expert system, Representation and organization of knowledge, Basic characteristics, Types of problems handled by expert systems, Case study of PROSPECTOR.

2. **Expert System Tools**
   Techniques of knowledge representation in expert systems, knowledge engineering, System-building aids, support facilities, stages in the development of expert systems.

3. **Building an Expert System**
   Expert system development, Selection of tool, Acquiring knowledge, Building process

4. **Problems with Expert Systems**
   Difficulties, common pitfalls in planning, Dealing with domain expert, Difficulties during development.

**Note:** At least one questions will be set from each unit.

**BOOKS**
Security and Cryptography  
COT-480  
(Open Elective II – For branches Co, IT, EC)

L T Theory: 75  
3 1 Sessionals: 50

1. Traditional Cryptography  
Cryptoanalysis, substitution and transposition ciphers, Cryptographic principles, secret-key algorithms: DES, DES chaining, Breaking DES, IDEA, Differential and Linear cryptoanalysis  
Public-key algorithms: RSA, Knapsack

2. Authentication protocols  
KDC protocol, shared secret key, Diffle-Hellman key exchange, Needham-Schroeder protocol, using Kerberos, interlock protocol, digital signatures – secret key and public key signatures, DSS, message digest, MD5 and secure hash algorithms

3. Computer Security Mechanisms  
Role of different security mechanisms, passwords – technology and administration, principles of database system security, epidemic of viruses: types of viruses, study of different virus codes, means of spread, prevention from virus, life cycle of a virus, immunization, Trojan horse and bombs with examples, writing antivirus/trojan codes.

4. Network Security  
Basics, security functions, preventing loss and damage, securing local area network – authorization, security plan and policy, securing enterprise network – setting priorities, security plans, securing network components, hardware security, levels of access control and authorization.

BOOKS  