UNIVERSITY GRANTS COMMISSION
NET BUREAU

Code No. : 87

Subject : COMPUTER SCIENCE AND APPLICATIONS
SYLLABUS AND SAMPLE QUESTIONS

Note :
There will be two question papers, Paper–II and Paper–III (Part–A & B). Paper–II will cover 50 Objective Type Questions (Multiple choice, Matching type, True/False, Assertion-Reasoning type) carrying 100 marks. Paper–III will have two Parts–A and B; Paper–III (A) will have 10 short essay type questions (300 words) carrying 16 marks each. In it there will be one question with internal choice from each unit (i.e., 10 questions from 10 units; Total marks will be 160). Paper–III (B) will be compulsory and there will be one question from each of the Electives. The candidate will attempt only one question (one elective only in 800 words) carrying 40 marks. Total marks of Paper–III will be 200.

PAPER–II

1. Discrete Structures


Computability : Models of computation—Finite Automata, Pushdown Automata, Non-determinism and NFA, DPDA and PDAs and Languages accepted by these structures. Grammars, Languages, Non-computability and Examples of non-computable problems.


Groups : Finite fields and Error correcting/detecting codes.
2. Computer Arithmetic

Propositional (Boolean) Logic, Predicate Logic, Well-formed-formulae (WFF), Satisfiability and Tautology.


Representation of Integers: Octal, Hex, Decimal, and Binary. 2's complement and 1's complement arithmetic. Floating point representation.

3. Programming in C and C++

Programming in C: Elements of C—Tokens, identifiers, data types in C. Control structures in C. Sequence, selection and iteration(s). Structured data types in C—arrays, struct, union, string, and pointers.


4. Relational Database Design and SQL

E-R diagrams and their transformation to relational design, normalization—1NF, 2NF, 3NF, BCNF and 4NF. Limitations of 4NF and BCNF.

SQL: Data Definition Language (DDL), Data Manipulation Language (DML), Data Control Language (DCL) commands. Database objects like—Views, indexes, sequences, synonyms, data dictionary.

5. Data and File structures

Data, Information, Definition of data structure. Arrays, stacks, queues, linked lists, trees, graphs, priority queues and heaps.


6. Computer Networks

Network fundamentals: Local Area Networks (LAN), Metropolitan Area Networks (MAN), Wide Area Networks (WAN), Wireless Networks, Inter Networks.

Reference Models: The OSI model, TCP/IP model.

Internetworking: Switch/Hub, Bridge, Router, Gateways, Concatenated virtual circuits, Tunnelling, Fragmentation, Firewalls.


7. System Software and Compilers


Loading, linking, relocation, program relocatability. Linkage editing.

Text editors. Programming Environments. Debuggers and program generators.

Compilation and Interpretation. Bootstrap compilers. Phases of compilation process. Lexical analysis. Lex package on Unix system.

Context free grammars. Parsing and parse trees. Representation of parse (derivation) trees as rightmost and leftmost derivations. Bottom up parsers—shift-reduce, operator precedence, and LR. YACC package on Unix system.


8. Operating Systems (with Case Study of Unix)

Main functions of operating systems. Multiprogramming, multiprocessing, and multitasking.

Memory Management: Virtual memory, paging, fragmentation.

Concurrent Processing: Mutual exclusion. Critical regions, lock and unlock.

UNIX

The Unix System: File system, process management, bourne shell, shell
variables, command line programming.

Filters and Commands: Pr, head, tail, cut, paste, sort, uniq, tr, join,
etc., grep, egrep, fgrep, etc., sed, awk, etc.

System Calls (like): Creat, open, close, read, write, isseek, link, unlink,
stat, fstat, umask, chmod, exec, fork, wait, system.

9. Software Engineering

System Development Life Cycle (SDLC): Steps, Water fall model, Prototypes,
Spiral model.

Software Metrics: Software Project Management.

Software Design: System design, detailed design, function oriented design,
object oriented design, user interface design. Design level metrics.

Clean room approach, software reengineering.

10. Current Trends and Technologies

The topics of current interest in Computer Science and Computer Applications
shall be covered. The experts shall use their judgement from time to time to
include the topics of popular interest, which are expected to be known for an
application development software professional, currently, they include:

Parallel Computing

Parallel virtual machine (pvm) and message passing interface (mpi)
libraries and calls. Advanced architectures. Today's fastest
computers.

Mobile Computing

Mobile connectivity—Cells, Framework, wireless delivery technology
and switching methods, mobile information access devices, mobile
data internetworking standards, cellular data communication
protocols, mobile computing applications. Mobile
databases—protocols, scope, tools and technology. M-business.

E-Technologies

Electronic Commerce: Framework, Media Convergence of
Applications, Consumer Applications, Organisation Applications.
Electronic Payment Systems: Digital Token, Smart Cards, Credit Cards, Risks in Electronic Payment System, Designing Electronic Payment Systems.


Software Agents: Characteristics and Properties of Agents, Technology behind Software Agents (Applets, Browsers and Software Agents)


Main concepts in Geographical Information System (GIS), E-cash, E-Business, ERP packages.

Data Warehousing: Data Warehouse environment, architecture of a data warehouse methodology, analysis, design, construction and administration.

Data Mining: Extracting models and patterns from large databases, data mining techniques, classification, regression, clustering, summarization, dependency modelling, link analysis, sequencing analysis, mining scientific and business data.

Windows Programming

Introduction to Windows programming—Win32, Microsoft Foundation Classes (MFC), Documents and views, Resources, Message handling in windows.

Simple Applications (in windows)

Scrolling, splitting views, docking toolbars, status bars, common dialogs.

Advanced Windows Programming

Multiple Document Interface (MDI), Multithreading, Object linking and Embedding (OLE). Active X controls. Active Template Library (ATL). Network programming.
Unit—I

Combinational Circuit Design, Sequential Circuit Design, Hardwired and Microprogrammed processor design, Instruction formats, Addressing modes, Memory types and organisation, Interfacing peripheral devices, Interrupts.

Microprocessor architecture, Instruction set and Programming (8085, P-III/P-IV), Microprocessor applications.

Unit—II

Database Concepts, ER diagrams, Data Models, Design of Relational Database, Normalisation, SQL and QBE, Query Processing and Optimisation, Centralised and Distributed Database, Security, Concurrency and Recovery in Centralised and Distributed Database Systems, Object Oriented Database Management Systems (Concepts, Composite objects, Integration with RDBMS applications), ORACLE.

Unit—III

Display systems, Input devices, 2D Geometry, Graphic operations, 3D Graphics, Animation, Graphic standard, Applications.


Unit—IV

Programming language concepts, paradigms and models.


Principles, classes, inheritance, class hierarchies, polymorphism, dynamic binding, reference semantics and their implementation.

Principles, functions, lists, types and polymorphisms, higher order functions, lazy evaluation, equations and pattern matching.

Principles, horn clauses and their execution, logical variables, relations, data structures, controlling the search order, program development in prolog, implementation of prolog, example programs in prolog.
Principles of parallelism, coroutines, communication and execution. Parallel Virtual Machine (PVM) and Message Passing Interface (MPI) routines and calls. Parallel programs in PVM paradigm as well as MPI paradigm for simple problems like matrix multiplication.

Preconditions, post-conditions, axiomatic approach for semantics, correctness, denotational semantics.

Compiler structure, compiler construction tools, compilation phases.

Finite Automata, Pushdown Automata. Non-determinism and NFA, DPDA, and PDAs and languages accepted by these structures. Grammars, Languages—types of grammars—type 0, type 1, type 2, and type 3. The relationship between types of grammars, and finite machines. Pushdown automata and Context Free Grammars. Lexical Analysis—regular expressions and regular languages. LEX package on Unix. Conversion of NFA to DFA. Minimizing the number of states in a DFA. Compilation and Interpretation. Bootstrap compilers.


Unit—V

Analog and Digital transmission, Asynchronous and Synchronous transmission, Transmission media, Multiplexing and Concentration, Switching techniques, Polling.


Unit—VI

Definition, Simple and Composite structures, Arrays, Lists, Stacks queues, Priority queues, Binary trees, B-trees, Graphs.

Unit—VII

Object, messages, classes, encapsulation, inheritance, polymorphism, aggregation, abstract classes, generalization as extension and restriction. Object oriented design. Multiple inheritance, metadata.

HTML, DHTML, XML, Scripting, Java, Servelets, Applets.

Unit—VIII

Software development models, Requirement analysis and specifications, Software design, Programming techniques and tools, Software validation and quality assurance techniques, Software maintenance and advanced concepts, Software management.

Unit—IX

Introduction, Memory management, Support for concurrent process, Scheduling, System deadlock, Multiprogramming system, I/O management, Distributed operating systems, Study of Unix and Windows NT.

Unit—X

Definitions, AI approach for solving problems.

Automated Reasoning with propositional logic and predicate logic—fundamental proof procedure, refutation, resolution, refinements to resolution (ordering/pruning/restiction strategies).

State space representation of problems, bounding functions, breadth first, depth first, A, A*, AO*, etc. Performance comparison of various search techniques.

Frames, scripts, semantic nets, production systems, procedural representations. Prolog programming.

Components of an expert system, Knowledge representation and Acquisition techniques, Building expert system and Shell.

RTNs, ATNs, Parsing of Ambiguous CFGs. Tree Adjoining Grammars (TAGs).

Systems approach to planning, Designing, Development, Implementation and Evaluation of MIS.

Decision-making processes, evaluation of DSS, Group decision support system and case studies, Adaptive design approach to DSS development, Cognitive style in DSS, Integrating expert and Decision support systems.
Elective—I

Theory of Computation: Formal language, Need for formal computational models, Non-computational problems, diagonal argument and Russel’s paradox.

Deterministic Finite Automaton (DFA), Non-deterministic Finite Automaton (NFA), Regular languages and regular sets, Equivalence of DFA and NFA. Minimizing the number of states of a DFA. Non-regular languages, and Pumping lemma.

Pushdown Automaton (PDA), Deterministic Pushdown Automaton (DPDA), Non-equivalence of PDA and DPDA.

Context free Grammars: Greibach Normal Form (GNF) and Chomsky Normal Form (CNF), Ambiguity, Parse Tree Representation of Derivations. Equivalence of PDA’s and CFG’s. Parsing techniques for parsing of general CFG’s—Early’s, Cook-Kassami-Younger (CKY), and Tomita’s parsing.

Linear Bounded Automata (LBA): Power of LBA. Closure properties.


Chomsky Hierarchy of languages: Recursive and recursively-enumerable languages.

Elective—II


Variable Length Codes: Prefix Codes, Huffman Codes, Lempel-Ziv (LZ) Codes. Optimality of these codes. Information content of these codes.

Error Correcting and Detecting Codes: Finite fields, Hamming distance, Bounds of codes, Linear (Parity Check) codes, Parity check matrix, Generator matrix, Decoding of linear codes, Hamming codes.


Data Compression Techniques: Representation and compression of text, sound, picture, and video files (based on the JPEG and MPEG standards).
Elective—III


Complexity of simplex algorithm(s). Exponential behaviour of simplex.

*Ellipsoid method* and Karmakar's method for solving LPPs. Solving simple LPPs through these methods. Comparison of complexity of these methods.

*Assignment and Transportation Problems*: Simple algorithms like Hungarian method, etc.


Elective—IV


Elective—V

*Unix*: Operating System, Structure of Unix Operating System, Unix Commands, Interfacing with Unix, Editors and Compilers for Unix, LEX and YACC, File system, System calls, Filters, Shell programming.

*Windows*: Windows environment, Unicode, Documents and Views, Drawing in a window, Message handling, Scrolling and Splitting views, Docking toolbars and Status bars, Common dialogs and Controls, MDI, Multithreading, OLE, Active X controls, ATL, Database access, Network programming.
SAMPLE QUESTIONS

PAPER–II

1. Which among the following is not a structured data type in C?
   (A) Union
   (B) Pointer
   (C) String
   (D) Boolean

2. A B-tree of order $m$ is an $m$-way search tree with
   (A) all leaves of the tree on the same level
   (B) each node, except for root and leaves, having less than $\frac{m}{2}$ subtrees
   (C) the root of the tree having more than $m$ subtrees
   (D) all its leaves connected to form a linked list

3. A browsor is a software tool that helps
   (A) linking of application program modules
   (B) viewing of application information
   (C) developing application programs
   (D) debugging of application software

PAPER–III (A)

1. Distinguish between hardwired and microprogrammed processor designs and compare their cost-performance qualities.
   Or
   Draw an interface diagram, indicating handshake signals, for a programmable peripheral interface device connected to the system bus.

PAPER–III (B)

11. Explain with suitable examples the Huffman coding scheme for producing variable length code. Describe the basis for such coding scheme.
   Or
   State and interpret the Shannon's theorem of channel capacity in a noisy channel. Discuss the effect of noise on information transmission rate.

***

87  11