Syllabus of LPUNEST Entrance Exam for Ph.D in Computer Science and Engineering

Section 1:

Discrete Structure:

- Sets, Relations, Functions, Pigeonhole Principle, Inclusion -Exclusion Principle, Equivalence and Partial orderings, Elementary counting techniques, Probability. Measures for Information and total information.
- Computability: Model of computation-Finite Automata, Pushdown Automata, Non-determinism and NFS, DPDA and PDAs and languages accepted by these structure, Grammars languages, Non computability and Example of non computable problems, Turing Machine Model, Halting Problem of Turing Machine, Decidable and Undecidable languages,Recursively Enumerable Language.
- Graph: Definitions, walks, paths, trails, connected graphs, regular and bipartite graphs, cycles and circuits, tree and rooted tree, Spanning trees, Eccentricity of a vertex radius and diameter of a graph, Central graphs, and centers of a tree, Hamiltonian and Eulerian graphs, and planar graphs.
- Groups: Finite fields and error correcting/detecting codes.

Computer Arithmetic:

- Proportional (Boolean Logic), Predicate Logic, Well formed formula (WFF), Satisfactory and Tautology.
- Logic Families: TTL, ECL and C-MOS gates, Boolean algebra and Minimization of Boolean functions, Flip-Flop-types, race conditions and comparison, Design of combinational and sequential circuits, Machine instructions and addressing modes, Instruction pipelining.
- Representation of Integer: Octal, Hex, Decimal and Binary, 2's Complement and 1's complement arithmetic, Floating point representation.

Section 2:

Programming in C and C++:

- Programming in C: Element of C-Tokens, Identifiers, Data types in C, Control structure in C, Sequence selection and iteration, Structured data types in C-arrays, struct, union, string and pointers.
- O-O Programming concepts: Classes, Object, installation, Inheritance, Polymorphism and overloading.

- C++ Programming: Element of C++ Tokens, identifiers, Variable and constants, Data types, Operators, Control Statements, Functions parameter passing, Class and objects, Constructors and destructors, overloading inheritance, Templates , Exception handling, File Handling.

Relational Database Design and SQL:

- E-R Diagram and their transformation to relation design, Normalization -INF, 2NF,3NF, 4NF and BCNF, Limitations of 4NF and BCNF, Transactions and concurrency control.
- SQL: Data definition language (DDL) Data manipulation language (DML), Data control language (DCL) commands. Database objects like- Views, indexes, sequences, synonyms, data dictionary.

Section 3:

Data and File Structure:

- Data information, Definition of data structure, Arrays, Stacks, Queues, Linked lists, Trees, Graphs, Priority queues, and heaps, Searching and Sorting Techniques, Algorithm Notation & Complexity analysis, NP Completeness.
- File Structure: Fields, records, and files. Sequential, direct, index sequential and relative files. Hashing, inverted lists and multi lists, B trees and B+ trees.

Computer Network

- Network Fundaments: Local area Networks (LAN), Metropolitan Area Networks (MAN), Wide area Networks (WAN), Wireless Networks, Ethernet.
- Reference models: The OSI model, TCP/IP Model, Protocols at each layer.
- Data Communication: Channel capacity, Transmission media-Twisted pair, Coaxial cables, Fiber optic cables, Wireless transmission - radio, microwave, infrared and millimeter waves, Light wave transmission, telephones-local loop, trunks, multiplexing, switching, ATM, High speed LANS, Cellular radio, Communication satellite- Geosynchronous and low orbit, Line coding techniques.
- Internetworking and Addressing: Switch/hub, bridge, router, gateways, concatenated virtual circuits, tunneling, fragmentation, IPv4 and IPv6 addressing, NAT.
- Routing: Virtual circuits and datagrams, Routing algorithms, Congestion control.
- Network security and Application Layer Protocols: Cryptography-public key and Private key, Hash based algorithms, Digital signatures and certificates, domain name system(DNS)- Electronic mail and world wide web(WWW).The DNS, resource records, Name servers, email architecture and servers.

Section 4:

System Software:

- Assemble language fundamentals (8085 based assembly language programming), Assemblers-2 pass and single pass, Macros and Macro processors.
- Loading, linking, relocation, program relocatability, linkage editing.
- Text editors, programming environments, Debuggers and program generators.
- Compilation and interpretation, bootstrap compilers, Phase of Compilation process, Lexical analysis, Lex package on UNIX System
- Context free grammars, parsing and parse tree, representation of parse trees as leftmost and rightmost derivations. Bottom up parsers-shift reduce, operator precedence and LR, YACC package on UNIX system.
- Topdown parsers-left recursion and its removal, Recursive descent parser, Predictive parser, Intermediate codes- Quadruples, Triples, Intermediate code generation, Code generation, Code optimization.

Operating System:

- Main function of operating system, Multiprogramming, multiprocessing and multitasking.
- Memory management: Virtual memory, Paging, Fragmentation.
- Concurrent processing: mutual exclusion, critical regions, lock and unlock, threads.
- Scheduling: CPU Scheduling, Disk scheduling, I/O Scheduling, resource scheduling, Deadlock and scheduling algorithms. Banker's Algorithm for deadlock handling.
- 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.
- Systems calls(like): Creat, open, close, read, write, iseek, link, unlink, stat, fstat, unmask, chmod, exec, fork, wait, system.

Section 5:

Software engineering:

- System development cycle (SDCL): 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.
- **Effort Estimation:** Empirical and heuristic estimation techniques, COCOMO.
- Coding and testing: testing level metrics, Software quality and reliability, Clean room approach, Software reengineering.
- **Software Testing:** Levels of testing, black-box and white-box testing techniques, Test case and test suit.

Artificial Intelligence:

- AI Basics: Heuristic search techniques, Supervised and Unsupervised learning, Production systems, Proposition Logic, FOPL.
- Neural Networks: Perceptron model, Activation Functions, Backpropagation, Hopfield network, Multilayer neural network.
- Fuzzy Systems: Fuzzy set, Fuzzy relations, Fuzzy functions, Fuzzy reasoning.

Section 6:

Algorithms and Analysis:

- Analysis of algorithms: Time and space complexity, Asymptotic notations, Recursion, recurrences and methods of solving recurrences.
- Sorting and searching algorithms: Bubble sort, Selection sort, Insertion sort, Shell sort, Quick sort, Radix sort, Merge sort. Linear and Binary search.
- Algorithm design techniques: Divide and Conquer, Brute force, Greedy method, Dynamic programming, Back tracking, Branch and Bound.
- Complexity Classes :Concepts of Complexity Classes- P, NP, NP-hard, NP-complete Problems

Current Trends and Technologies:

- Cloud Computing: Basic Cloud Types, Service and Deployment Model, High Availability and Resiliency, Serverless , Cloud Storage Techniques.
- Cyber Security: CIA Triad, Vulnerability, Threats, Malware Types, Password security, WiFi security, Phishing.
- Machine Learning: Confusion matrix, Linear regression, Decision tree, SVM, Ensemble learning techniques.