Dept. of Computer Science & Engineering
IIT Kharagpur
Syllabus for MS Comprehensive Viva-voce

The MS comprehensive viva-voce will test the student's competency in six basic areas of CSE, as well as in specific subjects the student has undertaken during his/her studies for the MS degree. The six basic areas tested and their syllabus are given below; the syllabus for the subjects taken by the students will follow the published syllabus of the individual subjects.

A. Discrete Mathematics

Proof Techniques: Direct proofs, proof by contraposition, proof by contradiction, proof by cases, existence proofs, Proofs by mathematical induction.

Sets: Subsets, set equality, set operations: complement, union, intersection, set difference, Powersets.

Functions: injective, surjective, bijective functions, Composition of functions, Inverse of a function.

Relations and their properties, Equivalence relations, Partial Orderings.

Counting, Principle of inclusion-exclusion, Pigeonhole principle, Permutations and Combinations.

Recurrence Relations: Linear recurrence relations, divide and conquer recurrence relations.

Discrete probability, Probability of combinations of events, Conditional probability, Independence, Bayes Theorem, Expected values.

Graph Theory: Basic terminologies, degree, connected components, tree

B. Algorithms

Introduction: The order notations, worst-case and average-case performance of algorithms, notions of lower bounds on problem complexity and optimality of algorithms.

Data structures: Stacks, queues, trees, graphs, binary search trees, height balancing – red-black trees, heaps and priority queues, hash tables.

Searching and sorting: Linear and binary search. Bubble sort, insertion sort, selection sort, merge sort, quick sort, heap sort, counting sort.

Algorithm design techniques: Divide-and-conquer, greedy, and dynamic programming algorithms. Lower bounds and optimal algorithms.

Graph algorithms: Preorder, inorder and postorder traversal of trees, BFS and DFS, topological sort. Minimum spanning trees (Kruskal's and Prim's algorithms), Shortest path (Dijkstra and Floyd-Warshal algorithms).

NP-completeness: The complexity classes P and NP, polynomial-time reductions, NP-hard and NP-complete problems.

C. Formal Languages and Automata Theory

Introduction: Alphabets, Strings, Languages, Grammars and Recognizers, Decision problems over languages

Regular languages (RLs), Regular Grammars and Regular Expressions, DFAs, NFAs and their equivalence with Regular Grammars and Regular Expressions, Converting NFA to DFA, Proving Languages not to be Regular -- pumping lemma for RLs, Closure of RLs under Boolean operations.

Context Free Languages (CFLs), CFGs, Parse trees, Ambiguities in Grammars and Languages, Pushdown Automata (PDAs).

Turing Machines -- Basic machine, Universal Turing machine and notions of undecidability

D. Computer Organization and Architecture

Instruction set architecture: instruction types, Instruction formats, addressing modes.

Arithmetic: representation of fixed and floating-point numbers, 2's complement arithmetic.

Control unit: Organization of a CPU, control and data paths, micro-operations, register-transfer level specifications.

Memory system: Typical signal lines in a ROM and RAM, building memory subsystems using smaller modules. Concept of memory hierarchy, cache memory, cache performance, cache-main memory mapping.

Input-output systems: Programmed I/O, Interrupt-driven I/O, polling and vectored interrupt, basic concept of DMA transfer.

Basic concepts of pipeline

E. Operating Systems

Basic components of an operating system.

Processes and Interprocess communiccation: basic concepts, process state transition diagram, context switch, process synchronization problem, semaphores - definition and implementation.

Process scheduling: preemptive and non-preemptive scheduling, FCFS, SJF and round-robin scheduling.

Memory management: logical and physical addresses, paging and virtual memory, page table, TLB.

Hardware requirements: privileged mode and privileged instructions, handling of hardware and software interrupts.

F. Programming

C programming: datatypes, variables, assignment statements, iteration and control statements, arrays, structures, functions and recursion, pointers

User-defined data types: stack, queue, binary tree, binary search tree

Representation of graphs