SOFTWARE ENGINEERING EXAMINATIONS

GROUP A

COMPULSORY EXAMINATIONS (SEVEN REQUIRED)

19-Soft-A1 Algorithms & Data Structures
Fundamental data structures and their associated algorithms. Stacks and queues, trees, tables, lists, arrays, strings, sets; files and access methods. B-trees, multi-key organizations. Searching. Sorting. Algorithm design techniques, such as divide and conquer, the greedy method, balancing, dynamic programming. Algorithms related to set operations, Graphs, graph algorithms: depth-first and breadth-first search, minimum spanning tree, shortest path. Empirical and theoretical measures of the efficiency of algorithms. Complexity analysis. Hard problems, NP-completeness, and intractable problems.

19-Soft-A2 Computer Architecture and Operating Systems
Computer Architecture basics, including Boolean algebra, gates, combinational and sequential logic, machine-level representation of data; machine organization, assembly/machine language programming; memory organization, caches, heaps, stacks; serial and parallel I/O, interrupts, bus protocols, and direct memory access (DMA). Operating System basics, including concurrency, process scheduling, memory management; protection, access, and authentication; linking and loading; virtual machines.

19-Soft-A3 Software Design

19-Soft-A4 Real-Time Systems

19-Soft-A5 Requirements and Specifications
Elicitation sources and techniques. Modelling paradigms, including information modelling, behavioural modelling, domain modelling, functional modelling, constraint modelling. Quality requirements (e.g., performance, usability, reliability, maintainability); expressing quality requirements so that they are testable. Prioritization, trade-off analysis, negotiation, risk analysis, and impact analysis. Requirements management, consistency management, interaction analysis, traceability. Requirements documentation (e.g., use cases) and specification languages. Validation, reviews and inspections, prototyping, validating non-functional requirements. Acceptance test design.

19-Soft-A6 Software Quality Assurance
Validation and verification concepts, software lifecycle and application of validation and verification, software quality assurance processes. Definitions of software product quality, quality characteristics,
engineering quality definitions, specifications. Definition and classifications of software defects, fitness for use and customer quality definitions. Software costs, quality costs and economics. Reviews, walkthroughs and inspections. Unit (Module/Package) level testing, subsystem/integration testing, regression testing, state based testing, traditional functional testing, logical testing/analysis, OO testing considerations (polymorphism and inheritance). Safety/failure analysis and testing.

19-Soft-A7 Software Development Process

GROUP B
OPTIONAL EXAMINATIONS (THREE REQUIRED)

19-Soft-B1 Advanced Software Design
Software design paradigms: object-oriented, service-oriented, component-based, agent-based, functional programming, client-server (including protocols such as REST), virtualization. Distributed component-based frameworks and systems. Design patterns. Model-driven design of software. Software architecture. Architecture representation.

19-Soft-B2 User interface

19-Soft-B3 Security
Security risks, threats, and vulnerabilities. Confidentiality, integrity, and privacy. Cryptography, access control, assurance, accountability. Engineering of secure systems, architectural approaches (e.g., confinement, virtual machines, trusted computing). Analysis techniques (e.g., static analysis and testing, model checking). Implications on human interface design and usability.

19-Soft-B4 Dependable systems
19-Soft-B5 Software Modeling & Verification (Formal Methods)
Mathematical modelling of software, including topics such as programming logics, process algebras, model based specification, object constraint languages, and algebraic specification. Mathematical reasoning using such models, including proofs of program correctness. Tools for static checking of the correctness of software relative to its specification.

19-Soft-B6 Software Project Management

19-Soft-B7 Reverse Engineering, Maintenance & Evolution

19-Soft-B8 Distributed Systems

19-Soft-B9 Parallel Computing

19-Soft-B10 Networking and Communications

19-Soft-B11 Process Control Systems
Discrete time models of continuous physical phenomena. Z-transform and transfer functions. Time domain and frequency domain response of first, second and higher order systems. Stability and feedback compensation. Steady state error and proportional, integral and derivative (PID) control. Compensator design using Nyquist criterion and frequency domain design. Sampling theorem, aliasing,
Computer control interfacing.

19-Soft-B12 Engineering Computation: Numerics
Representation of numbers and floating-point round-off. Caveats of computations with floating point.
Linear systems: direct and iterative methods, conditioning, structured systems. Zeros of functions.

19-Soft-B13 Performance Analysis & Simulation
Basic techniques of system performance evaluation. Specific topics include: measurement methods and tools, experimental design and analysis, modeling (including queueing and network of queueing systems), discrete event simulation, verification and validation of simulation models, analysis of simulation output, statistical methods (comparing systems using sample data, hypothesis testing and confidence measures).

19-Soft-B14 Safety Critical Systems
Safety and hazard analysis. Use of software in safety related systems. Legal and ethical considerations.

19-Soft-B15 Artificial Intelligence

19-Soft-B17 Programming Languages, semantics and implementation

19-Soft-B18 Data Visualization

20-Soft-B19 Discrete Mathematics
Logic: propositional equivalences, predicates and quantifiers, sets, set operations, functions,