Syllabus for Part I of the Computer Science Ph.D.
Comprehensive Examination
Department of Computing and Software
McMaster University

CAS-2012-09

Area 1 - Computing Fundamentals

1. Fundamental Data Structures
   Arrays, lists, queues, stacks, trees, priority queues, balanced trees, sets. Abstraction and abstract data types.

2. Discrete mathematics
   Sets, functions, relations; partial and linear orders, lattices, boolean algebras; basic algebraic structures like monoids, rings; graphs and trees; sequences and series.

3. Combinatorics
   Basic counting principles; permutations and combinations; basic probability

4. Logic
   Syntax vs. semantics; languages, theories, models; propositional logic; first-order logic; higher-order logic; formal proof systems; inductive proofs; pre-post conditions; weakest pre-condition; loop invariants; program verification

5. Algorithms
   Sorting and searching algorithms; algorithm design schemes such as greedy algorithms, dynamic programming; graph and network algorithms; linear programming; recursion; algorithm complexity

6. Theory of computation
   Regular, context-free, context-sensitive, and recursively enumerable languages; finite automata; Church-Turing thesis and common models of computation such as recursive functions, Turing machines; complexity classes

7. Information security
   Confidentiality, authentication and integrity; defence mechanisms; cryptography; network security; secure communication protocols; security management

Area 2 - Computer Science

1. Programming languages
   BNF and other syntax-definition schemes; language paradigms: imperative, functional, logic programming; use of types; data structures; language mechanisms: input/output, modularity, object-oriented, exception handling; implementation issues: lazy vs. eager evaluation, garbage collection, parameter passing; program semantics and correctness

2. Scientific computation
   Number representations; floating-point number systems; rounding errors; underflow and overflow; roots of equations; interpolation; quadrature rules; systems of linear and non-linear equations; ordinary and partial differential equations; linear and non-linear least squares; minimization of functions; eigenvalue decomposition

3. Computer architecture
   Logic design; instruction sets; machine language; computer arithmetic; parallelism; memory hierarchy; multiprocessing; storage systems

4. Operating systems
   Role of an OS; OS interface; layering structure; processes and threads; synchronization and communication; scheduling; concurrency; memory management; files systems

5. Databases
   Structure of a DBMS; design principles; relational model; database processing: transactions, recovery and concurrency control, deadlock detection and avoidance, access control

6. Computer networks
   Physical networks; packet vs. circuit-switched networks; wide area vs. local area networks; internets and other virtual networks; TCP/IP protocol suite; TCP/IP and OSI layered network models; client/server model; common network services

7. Software design/development and specification
   Role of specifications; modularization, information hiding, and module interfaces; specification paradigms: pre- and postconditions, algebraic, logical; abstraction and refinement; informal and formal specification languages; specification development tools

8. Human-computer interaction
   HCI design principles; HCI hardware; evaluation of HCIs; data visualization