Institute for
Certification of
Computing
Professionals

Computing Science Exam Outline

1.0. Algorithms and Complexity

1.1. Basic Analysis
- Algorithmic behaviors
- Bounds

1.2. Algorithmic Strategies
- Brute force, greedy, divide & conquer, recursive backtracking, dynamic programming

1.3. Fundamental Data Structures and Algorithms
- Simple numeric algorithms
- Secquential & Binary Search
- Quadratic Sorts
- Hash tables, avoiding collisions
- Binary Search Trees, common notations
- Graphs and graph algorithms, represtiations, traversals

1.4. Basic Automata Computability and Complexity
- Finite state machines
- Regular expressions
- The halting problem

2.0. Computational Science

2.1. Introduction to Modeling and Simulation
- Models as abstractions
- Simulations as dynamic modeling
- Simulation techniques
- Validating models & presenting

3.0. Discrete Structures

3.1. Sets, Relations, and Functions
- Venn Diagrams
- union, intersection, complement
- Cartesian product
- power sets
- Reflexivity, symmetry, transitivity
- Equivalence relations, partial orders
- Surjections, injections, bijections
- Inverses
- composition

3.2. Basic Logic

- Propositional logic
- Logical connectives
- Truth tables
- Normal forms (Conjunctive and disjunctive)
- Validity of well-formed formula

3.3. Proof Techniques

- Notions of implication, equivalence, converse, inverse, contrapositive, negation, and contradiction
- The structure of mathematical proofs
- Direct proofs
- Disproving by counterexample
- Proof by contradiction
- Induction over natural numbers
- Structural induction
- Weak and strong induction (i.e., First and Second Principle of Induction)
- Recursive mathematical definitions

3.4. Basics of Counting

- Counting arguments: set cardinality, counting, sum, product rule, Inclusion-exclusion, arithmetic and geometric progressions
- The pigeonhole principle
- Permutations and combinations: Pascal’s identity, binomial theorem
- Solving recurrence relations (cross-reference: AL/Basic Analyses e.g. Fibonacci numbers, others
- Basic modular arithmetic

3.5. Graphs and Trees

- Trees: Properties, Traversal strategies
- Graphs: Undirected, Directed, & Weighted
- Spanning trees/forests

3.6. Discrete Probability

- Finite probability space, events
- Axioms of probability and probability measures
- Conditional probability, Bayes’ theorem
- Independence
- Integer random variables (Bernoulli, binomial)
- Expectation, including Linearity of Expectation

4.0. Graphics and Visual Computing

4.1. Fundamental Concepts

- Media applications: video, game, CADD, Visualization, VR
- Digitization: analog, resolution, visual perception limits, audio, print
- Standard APIs for construction of user interfaces
- Standard media formats (includes: lossless & lossy formats)

5.0. Human-Computer Interaction

5.1. Foundations

- Contexts: web, business, mobile, games
- User centered development: engagement, empirical, iterative
- Use evaluation: utility, efficiency, learnability, satisfaction
- Heuristics and usability testing
- Ergonomics: Physical capability, informed interaction, color
- Cognitive models: attention, perception, recognition, movement, memory, expectations vs outcomes
- 5Trade-offs: design, designers
- Accessibility: blind, motion-impaired, audio, age-based

6.0. Information Assurance and Security

6.1. Foundational Concepts in Security

- CIA: Confidentiality, Integrity, Availability
- Risks, threats, vulnerability, attack sectors
- Authentication, authorization, access control
- Trust, Trustworthiness
- Ethics (responsible disclosure)

6.2. Principles of Secure Design

- Least privilege and isolation (applications, memory, equipment)
- Fail-safe defaults (coding, idioms/patterns, application correctness)
- Open design (large existing code-base)
- End-to-end security (errors over increasing communications distance)
- Defense in depth (defensive programming, layered defense)
- 6Security by design (encryption, decryption – data and applications)
- 6Security versus other design goals

6.3. Defensive Programming

- Input validation and sanitization
- Programming language selection and type-safe languages
- Programming Examples: Buffer overflows, integer errors, SQL injection, XSS vulnerability
- Race conditions: parallel situations: programming, processing, distributed and shared memory.
- Correct handling of exceptions and unexpected behaviors

7.0 Information Management

7.1. Information Management Concepts

- IS as socio-technical systems
- Information storage and retrieval
- Information capture and representation
- Human needs: searching, retrieval, linking, browsing, navigation

8.0. Networking and Communications

8.1. Introduction

- Internet: organization
- Switching techniques: circuit, packet
- Physical components: hosts, routers, switches, ISPs, wireless, LAN, access point, firewalls
- Layering principles: encapsulation, multiplexing
- Roles of the communications layers: application, transport, network, datalink, physical

8.2. Networked Applications

- Naming and address schemes: DNS, IP, Uniform resource identifiers)
- Distributed Applications (cloud, client/server, peer-to-peer)
- HTTP application layer
- Multiplexing, TCP, UDP
- Socket APIs

9.0. Operating Systems

9.1. Overview of Operating Systems

- Role and purpose
- Functionality
- Mechanisms to support client-server models, hand-held devices
- Design issues (efficiency, robustness, flexibility, portability, security, compatibility)
- Influences of security, networking, multimedia, windowing systems

9.2. Operating Systems Principles

- Structuring methods (monolithic, layered, modular, micro-kernel models)
- Abstractions, processes, and resources
- Application program interfaces (APIs)
- Evolution of hardware/software techniques and application needs
- Device organization
- Interrupts: methods and implementations
- User/system state and protection, transition to kernel mode

10.0. Parallel and Distributed Computing

10.1. Parallel Architecture

- Multiple simultaneous computations
- Goals: throughput versus concurrency; controlling access to shared resources
- Parallelism, communication, and coordination
  - Coordinating multiple simultaneous computations
  - Need for synchronization
- Programming errors
  - Data races
  - Higher-level
  - Deadlock, starvation

10.2. Parallel Decomposition

- Communication and coordination/synchronization
- Independence and partitioning

10.3. Communication and Coordination

- Shared Memory
- Consistency

11.0 Programming Languages

11.1. Object Oriented Programming

- Object-oriented design
  - Decomposition into objects carrying state and having behavior
  - Class-hierarchy design for modeling
- Definition of classes: fields, methods, and constructors
- Subclasses, inheritance, and method overriding
- Dynamic dispatch: definition of method-call

11.2. Functional Programming

- Effect-free programming
- Function calls
  - Immutable variables are
  - Aliased Data avoiding mutation
- Processing structured data
  - Associated language constructs - discriminated unions and pattern-matching
  - Functions defined over compound
- First-class functions (taking, returning, and storing functions)

11.3. Basic Type Systems

- Set a values and Operations
  - Primitive types (e.g., numbers, Booleans) o Compound types built from other types (e.g., records, unions, arrays, lists, functions, references) • Association of types to variables, arguments, results, and fields • Type safety and errors caused by using values inconsistently given their intended types • Goals and limitations of static typing o Eliminating some classes of errors without running the program o Undecidability means static analysis must conservatively approximate program behavior

12.0. Software Development Fundamentals

12.1. Algorithms and Design
12.2. Fundamental Programming Concepts
12.3. Fundamental Data Structures
12.4. Development Methods

13.0. Software Engineering

13.1. Software Processes
13.2. Requirements Engineering
13.3. Softare Design

14.0. Systems Fundamentals

14.1. Computational Paradigms
14.2. Cross-Layer Communications
14.3. State and State Machines
14.4. Parallelism
14.5. Evaluation

15.0. Social Issues and Professional Practice

15.1. Social Context
15.2. Analytical Tools
15.3. Professional Ethics
15.4. Intellectual Property
15.5. Privacy and Civil Liberties
15.6. Professional Communication
15.7. Sustainability