



# GRE<sup>®</sup>

Listening. Learning. Leading.<sup>®</sup>

We invite  
you to

**Take a  
Closer  
Look...**

For more information  
about this GRE Subject  
Test, contact the GRE  
Program:

Phone: 1-609-683-2002

Fax: 1-609-683-2040

E-mail: [gretests@ets.org](mailto:gretests@ets.org)

Educational Testing Service  
Rosedale Road  
Princeton, NJ 08541

## The GRE<sup>®</sup> Computer Science Test

Does your graduate department require or recommend that graduate applicants take the Computer Science Subject Test offered by the *Graduate Record Examinations<sup>®</sup>* Program?

This Subject Test can be very useful in distinguishing among candidates whose credentials are otherwise similar. The test measures undergraduate achievement and provides a common yardstick for comparing the qualifications of students from a variety of colleges and universities with different standards. Consider these factors:

### **Predictive validity**

Subject Test scores are a valid predictor of graduate school performance, as confirmed by a recent meta-analysis performed by independent researchers of over 1,700 studies containing validity data for GRE tests.\* This study showed that GRE Subject Tests are reliable predictors of a range of outcome measures, including first-year graduate grade-point average, cumulative graduate grade-point average, comprehensive examination scores, publication citation counts, and faculty ratings. For more information about the predictive validity of the GRE tests, visit [www.ets.org/gre/validity.html](http://www.ets.org/gre/validity.html).

### **Content that reflects today's curricula**

The test contains about 70 multiple-choice questions involving three major areas: software systems and methodology, computer organization and architecture, and theory and mathematical background. A more detailed list of test content areas can be found on the back of this sheet. Some questions are grouped in sets and based on such materials as diagrams, graphs, and program fragments. Additional information about the test and a full-length practice test are provided FREE with test registration and can be downloaded at [www.ets.org/gre/greprep](http://www.ets.org/gre/greprep).

### **Developed by leading educators in the field**

The content and scope of each edition of the test are specified and reviewed by a distinguished team of undergraduate and graduate faculty representing colleges and universities across the country.

\*Source: "A comprehensive meta-analysis of the predictive validity of the Graduate Record Examinations<sup>®</sup>: Implications for graduate student selection and performance." Kuncel, Nathan R.; Hezlett, Sarah A.; Ones, Deniz S., *Psychological Bulletin*, January 2001, Vol. 127(1), 162-181.

## Who develops the GRE Computer Science Test?

Individuals who serve or have recently served on the Committee of Examiners are faculty members from the following institutions:

Carnegie Mellon University  
Florida A&M University  
Harvard University  
James Madison University  
Princeton University  
University of Chicago  
University of Toronto  
University of Washington  
Washington University

Committee members are selected with the advice of the Association for Computing Machinery (ACM) and the Computer Society of the Institute of Electrical and Electronics Engineers (IEEE).

Test questions are written by committee members and by other subject-matter specialists from ETS and colleges and universities across the country.

75987-56062 • WEBPDF118  
I.N. 749431

## Test Content

### I. SOFTWARE SYSTEMS AND METHODOLOGY (40%)

- A. Data organization
  - 1. Data types
  - 2. Data structures and implementation techniques
- B. Program control and structure
  - 1. Iteration and recursion
  - 2. Procedures, functions, methods, and exception handlers
  - 3. Concurrency, communication, and synchronization
- C. Programming languages and notation
  - 1. Constructs for data organization and program control
  - 2. Scope, binding, and parameter passing
  - 3. Expression evaluation
- D. Software engineering
  - 1. Formal specifications and assertions
  - 2. Verification techniques
  - 3. Software development models, patterns, and tools
- E. Systems
  - 1. Compilers, interpreters, and run-time systems
  - 2. Operating systems, including resource management and protection/security
  - 3. Networking, Internet, and distributed systems
  - 4. Databases
  - 5. System analysis and development tools

### II. COMPUTER ORGANIZATION AND ARCHITECTURE (15%)

- A. Digital logic design
  - 1. Implementation of combinational and sequential circuits
  - 2. Optimization and analysis
- B. Processors and control units
  - 1. Instruction sets
  - 2. Computer arithmetic and number representation
  - 3. Register and ALU organization
  - 4. Data paths and control sequencing
- C. Memories and their hierarchies
  - 1. Performance, implementation, and management
  - 2. Cache, main and secondary storage
  - 3. Virtual memory, paging, and segmentation
- D. Networking and communications
  - 1. Interconnect structures (e.g., buses, switches, routers)
  - 2. I/O systems and protocols
  - 3. Synchronization
- E. High-performance architectures
  - 1. Pipelining superscalar and out-of-order execution processors
  - 2. Parallel and distributed architectures

### III. THEORY AND MATHEMATICAL BACKGROUND (40%)

- A. Algorithms and complexity
  - 1. Exact and asymptotic analysis of specific algorithms
  - 2. Algorithmic design techniques (e.g., greedy, dynamic programming, divide and conquer)
  - 3. Upper and lower bounds on the complexity of specific problems
  - 4. Computational complexity, including NP-completeness
- B. Automata and language theory
  - 1. Models of computation (finite automata, Turing machines)
  - 2. Formal languages and grammars (regular and context free)
  - 3. Decidability
- C. Discrete structures
  - 1. Mathematical logic
  - 2. Elementary combinatorics and graph theory
  - 3. Discrete probability, recurrence relations, and number theory

### IV. OTHER TOPICS (5%)

Example areas include numerical analysis, artificial intelligence, computer graphics, cryptography, security, and social issues.

**Note:** Students are assumed to have mathematical background in the areas of calculus and linear algebra as applied to computer science.



**GRE**

*Listening. Learning. Leading.*