MAJOR FIELD TESTS

Colleges and universities use the Major Field Tests to measure student academic achievement and growth and to assess the educational outcomes of their major programs. In addition, academic departments use the Major Field Tests to evaluate their curricula and to measure the progress of their students. The tests also provide students with an assessment of their own level of achievement within a field of study compared to that of students in their program and to national comparative data.

Background

Development of the Major Field Tests began in 1989, modeled on the development of the Graduate Record Examinations (GRE) Subject Tests. However, unlike the GRE Subject Tests, the Major Field Tests do not serve as a predictor of graduate school success, but are designed to measure the basic knowledge and understanding achieved by senior undergraduates in their major field of study. Each test is revised approximately every five years. Experienced teaching faculty members representing all the relevant areas of a discipline participate in determining test specifications, questions, and types of scores reported. ETS assessment experts subject each question to rigorous tests of sensitivity and reliability. In addition, every effort is made to include questions that assess the most common and most important topics and skills within each major field of study.

Test Content

The Major Field Tests are designed to assess mastery of concepts, principles, and knowledge expected of students at the conclusion of an academic major in specific subject areas. In addition to factual knowledge, the tests evaluate students' abilities to analyze and solve problems, understand relationships, and interpret material. The tests may contain questions that require interpretation of graphs, diagrams, and charts based on material related to the field. Academic departments may add up to 50 additional locally written questions to test areas of a discipline that may be unique to the department or institution.

Test Length

The tests are two-hour multiple-choice tests. The addition of optional, locally developed questions may require a longer testing period.

Test Administration

Departments or schools choose when and where to give the tests; however, the tests are normally administered during the senior year when students have completed the majority of courses in the major. Many institutions administer the tests as part of the requirements of a capstone course.

National Comparative Data

A Comparative Data Guide, published each year, contains tables of scale scores and percentiles for individual student scores, departmental mean scores, and any subscores or group assessment indicators that the tests may support. The tables of data are drawn from senior-level test takers at a large number of diverse institutions. More than 500 colleges and universities employ one or more of the Major Field Tests for student achievement and curriculum evaluation each year.

Scores

Major Field Test score reports are sent directly to the office within an institution that purchases them, such as a department chairperson, dean, or director of testing. Results of the tests are reported for the entire group of test takers, as well as for individual students. Overall student scores are reported on a scale of 120-200; subscores (which many of the tests include) are reported on a scale of 20-100. Another score reported for most of the tests is based on group-level achievement in subfields of the discipline. These "assessment indicators" report the average percent of a subset of test questions answered correctly by all students tested. On Major Field Tests, only correct answers are scored, so students are not penalized for omissions or guesses.

COMPUTER SCIENCE

COMPUTER SCIENCE (3XMF)

(Current form introduced in Spring 2002)

The Major Field Test in Computer Science consists of 60 multiple choice questions, some of which are grouped in sets and based on such materials as diagrams, graphs, and program fragments. The test is divided into four subareas with content distribution as follows:

I. **Programming Fundamentals** (26 percent of the questions)

A. Fundamental programming constructs and data structures
   1. Conditional and iterative control structures
   2. Basic data structures (primitive, arrays, records, pointers)
   3. Expression evaluation and statement execution

B. Problem-solving, algorithms, and recursion
   1. Problem-solving strategies (top-down, functional decomposition)
   2. Properties of algorithms
   3. Recursive procedures (Tower of Hanoi, generating permutations, divide and conquer)

C. Abstract data types
   1. Abstract programming interfaces and encapsulation
   2. Specific ADT structures (stacks, queues, symbol tables, graphs)

1) **Software Systems** (18 percent of the questions)

A. System management (5 percent)
   - Compilers and interpreters
   - Operating system, including resource management and protection/security

B. Concurrency (6 percent)
   - Communication and synchronization
   - Networking and distributed systems

C. System analysis and development (7 percent)
   - System development tools
   - System performance
   - Design and analysis (software engineering)

2) **Computer Organization and Architecture** (20 percent of the questions)

A. Logic design (5 percent)
   - Implementation of combinational and sequential circuits
   - Functional properties of digital integrated circuits
   - Processors and control units

B. Processors and control units (5 percent)
   - Instruction sets
   - Register and ALU organization
   - Control sequencing
   - Data paths

C. Memories and their hierarchies (3 percent)
   - Speed, capacity, cost
   - Cache, main, secondary storage
   - Virtual memory, paging, segmentation

D. Communication (4 percent)
   - Bus, switch, and network structures and protocols
   - I/O
   - Synchronization

E. High-performance architectures (3 percent)
   - Pipelining
   - Multiprocessors
   - Vector processors

3) **Theory and Computational Mathematics** (28 percent of the questions)

A. Automata and language theory (5 percent)
   - Models of computation (finite automata, pushdown automata, Turing machines)
   - Formal languages (regular languages, context-free languages)
   - Decidability

B. Analysis of algorithms and computational complexity (7 percent)
   - Exact or asymptotic analysis of the best, worst, or average case of the time and space complexity of specific algorithms'
   - Upper and lower bounds on the complexity of specific problems
   - NP-completeness

C. Correctness of programs (4 percent)
   - Formal specifications and assertions
   - Verification techniques

D. Discrete structures (12 percent)
   - Mathematical logic
   - Elementary combinatorics, including graph theory and counting arguments
   - Elementary discrete mathematics, including number theory, discrete probability, recurrence relations

4) **Special Topics** (12 percent of the questions)

A. Modeling and simulation
B. Data management
C. Artificial intelligence
D. Computer graphics
E. Data communications
F. Numerical algorithms and linear algebra
G. Parallel computers
H. Social, ethical, professional issues

Scores on the Computer Science Test are reported as follows:

Total Score
Reported for each student and summarized for the group.

Assessment Indicators
Reported for the group* only.

- Programming Methodology (13)
- Software Systems (11)
- Computer Organization and Architecture (12)
- Theory and Computational Mathematics (17)

Numbers in parentheses are approximate number of questions in each category.

*A minimum of five students is required for Assessment Indicators to be reported.