Find out how to prove — and improve — the effectiveness of your Computer Science program with the ETS® Major Field Tests.

Content Validity

The Major Field Test (MFT) in Computer Science, first administered in 1989, assesses mastery of concepts, principles and knowledge by graduating Computer Science students. To ensure fairness and content relevance, the test is revised approximately every four to five years.

Developed by Leading Educators in the Field

Experienced faculty members representing all the relevant areas of the discipline determine test specifications, questions and types of scores reported. ETS assessment experts subject each question to rigorous tests of sensitivity and reliability. Every effort is made to include questions that assess the most common and important topics and skills.

In addition to factual knowledge, the test evaluates students' abilities to analyze and solve problems, understand relationships and interpret material. Questions that require interpretation of graphs, diagrams and charts are included. Academic departments may add up to two subgroups and as many as 50 additional locally written questions to test areas of the discipline that may be unique to the department or institution.

National Comparative Data

A *Comparative Data Guide*, published each year, contains tables of scaled scores and percentiles for individual student scores, departmental mean scores and any subscores or group assessment indicators that the test may support. The tables of data are drawn from senior-level test takers at a large number of diverse institutions. Nearly 1,500 colleges and universities employ one or more of the Major Field Tests for student achievement and curriculum evaluation each year.

Who Develops the MFT in Computer Science?

Individuals who serve or recently have served on the Committee for the MFT in Computer Science are faculty members from the following institutions:

East Carolina University
George Fox University
**Indiana University-
    Purdue University**
Longwood University
Mt. St. Mary's University
Texas A&M University

For more information about the MFT in Computer Science:

Phone: **1-800-745-0269**
Email: **highered@ets.org**
Visit: **www.ets.org/mft**

Educational Testing Service
Rosedale Road
Princeton, NJ 08541

ETS® *Major Field Tests*

Listening. Learning. Leading.®

Test Content — Computer Science

The Major Field Test in Computer Science consists of 66 multiple-choice questions, some of which are grouped in sets and based on materials such as diagrams, graphs and program fragments. Programs can choose when and where to administer the tests. It is designed to take two hours and may be split into two sessions. This test must be given by a proctor. Mathematical operations do not require the use of a calculator. A table containing notations, conventions and definitions is available for reference; it is under "Exhibits" for online users and inside the test book for paper-and-pencil users.

The outline below shows the content areas covered on the test and the approximate distribution of questions among content areas.

The Test Outline

I. Discrete Structures (15–21%)

A. Functions, relations and sets
B. Basic logic
C. Proof techniques
D. Basics of counting and number theory
E. Graphs and trees
F. Discrete probability

II. Programming (21–27%)

A. Programming Fundamentals: fundamental programming constructs, basic algorithms and problem solving, fundamental data structures, recursion, event-driven programming, object-oriented programming
B. Programming Languages: features, paradigms, implementation techniques

III. Algorithms and Complexity (16–22%)

Advanced data structures and algorithms (including graph algorithms), algorithmic strategies, distributed algorithms, basic computability and complexity, automata theory and formal languages

IV. Systems (16–24%)

A. Architecture: digital logic and digital systems, machine level representation of data, assembly level machine organization, interfacing and communication
B. Operating Systems: operating system principles, concurrency, scheduling and dispatch, and memory management
C. Networking

V. Software Engineering (3–9%)

Software requirements, specifications, design, validation and management

VI. Information Management (3–8%)

Database systems and data modeling

VII. Other (3–8%)

Human-computer interaction, graphics, intelligent systems, social and professional issues, web computing and security

MFT Computer Science Pseudocode Statement

We currently do not use any specific programming languages in questions on the MFT Computer Science Exam. Instead, we use a simple pseudocode that we believe will be easily understood by any computer science student. See the examples below.

Example 1. Class declaration and object instantiation

```
class StudentInfo
    int studentID
    string name
end class StudentInfo

StudentInfo x ← new StudentInfo()
x.studentID ← 1234  // the value 1234 is assigned to x.studentID
x.name ← "John"
print ( x.name )
if ( x.name == "John" )
    print ( x.studentID )
end if
```

Example 2. The following procedure swaps the values of two parameters.

```
swap ( pass-by-reference int x, pass-by-reference int y )
    int temp ← x
    x ← y
    y ← temp
end swap
```

Example 3. SelectionSort

```
Preconditions:  A is an array of integers.
                The length of array A is n.
                The index of array A starts at 0.
int[] selectionSort ( pass-by-reference int[] A, int n )
    int min
    int j
    int i ← 0
    while ( i ≤ n - 1 )
        min ← i
        j ← i + 1
        while ( j ≤ n - 1 )
            if ( A[j] < A[min] )
                min ← j
            end if
            j ← j + 1
        end while
        if ( min ≠ i )
            swap ( A[min], A[i] )
        end if
        i ← i + 1
    end while
    return A  // returns the sorted array
end selectionSort
```

How scores for the Major Field Test in Computer Science are reported

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

Assessment Indicators — Reported for the group* only

– Programming and Software Engineering (22)
– Discrete Structures and Algorithms (24)
– Systems (Architecture, Operating Systems, Networking, Database) (17)

Numbers in parentheses are the approximate number of questions in each category
* A minimum of five (5) students is required for assessment indicators to be reported.