GRADUATE RECORD EXAMINATIONS®

Computer Science Test
Practice Book

This practice book contains

- one actual, full-length GRE® Computer Science Test
- test-taking strategies

Become familiar with

- test structure and content
- test instructions and answering procedures

Compare your practice test results with the performance of those who took the test at a GRE administration.

Visit GRE Online at www.ets.org/gre

Note to Test Takers: Keep this practice book until you receive your score report.
This book contains important information about scoring.

Table of Contents

Purpose of the GRE Subject Tests

The GRE Subject Tests are designed to help graduate school admission committees and fellowship sponsors assess the qualifications of applicants in specific fields of study. The tests also provide you with an assessment of your own qualifications.

Scores on the tests are intended to indicate knowledge of the subject matter emphasized in many undergraduate programs as preparation for graduate study. Because past achievement is usually a good indicator of future performance, the scores are helpful in predicting success in graduate study. Because the tests are standardized, the test scores permit comparison of students from different institutions with different undergraduate programs. For some Subject Tests, subscores are provided in addition to the total score; these subscores indicate the strengths and weaknesses of your preparation, and they may help you plan future studies.

The GRE Board recommends that scores on the Subject Tests be considered in conjunction with other relevant information about applicants. Because numerous factors influence success in graduate school, reliance on a single measure to predict success is not advisable. Other indicators of competence typically include undergraduate transcripts showing courses taken and grades earned, letters of recommendation, and GRE General Test scores. For information about the appropriate use of GRE scores, see the *GRE Guide to the Use of Scores* at **www.ets.org/gre/subject/scores/understand**.

Development of the Subject Tests

Each new edition of a Subject Test is developed by a committee of examiners composed of professors in the subject who are on undergraduate and graduate faculties in different types of institutions and in different regions of the United States and Canada. In selecting members for each committee, the GRE Program seeks the advice of appropriate professional associations in the subject.

The content and scope of each test are specified and reviewed periodically by the committee of examiners. Test questions are written by committee members and by other university faculty members who are subject-matter specialists. All questions proposed for the test are reviewed and revised by the committee and subject-matter specialists at ETS. The tests are assembled in accordance with the content specifications developed by the committee to ensure adequate coverage of the various aspects of the field and, at the same time, to prevent overemphasis on any single topic. The entire test is then reviewed and approved by the committee.

Subject-matter and measurement specialists on the ETS staff assist the committee, providing information and advice about methods of test construction and helping to prepare the questions and assemble the test. In addition, each test question is reviewed to eliminate language, symbols, or content considered potentially offensive, inappropriate for major subgroups of the test-taking population, or likely to perpetuate any negative attitude that may be conveyed to these subgroups.

Because of the diversity of undergraduate curricula, it is not possible for a single test to cover all the material you may have studied. The examiners, therefore, select questions that test the basic knowledge and skills most important for successful graduate study in the particular field. The committee keeps the test up-to-date by regularly developing new editions and revising existing editions. In this way, the test content remains current. In addition, curriculum surveys are conducted periodically to ensure that the content of a test reflects what is currently being taught in the undergraduate curriculum.

After a new edition of a Subject Test is first administered, examinees' responses to each test question are analyzed in a variety of ways to determine whether each question functioned as expected. These analyses may reveal that a question is ambiguous, requires knowledge beyond the scope of the test, or is inappropriate for the total group or a particular subgroup of examinees taking the test. Such questions are not used in computing scores.

Following this analysis, the new test edition is equated to an existing test edition. In the equating process, statistical methods are used to assess the difficulty of the new test. Then scores are adjusted so that examinees who took a more difficult edition of the test are not penalized, and examinees who took an easier edition of the test do not have an advantage. Variations in the number of questions in the different editions of the test are also taken into account in this process.

Scores on the Subject Tests are reported as three-digit scaled scores with the third digit always zero. The maximum possible range for all Subject Test total scores is from 200 to 990. The actual range of scores for a particular Subject Test, however, may be smaller. For Subject Tests that report subscores, the maximum possible range is 20 to 99; however, the actual range of subscores for any test or test edition may be smaller. Subject Test score interpretive information is provided in *Interpreting Your GRE Scores*, which you will receive with your GRE score report. This publication is also available at **www.ets.org/gre/subject/scores/understand**.

Content of the Computer Science Test

The test consists of approximately 70 multiple-choice questions, some of which are grouped in sets and based on such materials as diagrams, graphs and program fragments.

The approximate distribution of questions in each edition of the test according to content categories is indicated by the following outline.

The percentages given are approximate; actual percentages will vary slightly from one edition of the test to another.

I. SOFTWARE SYSTEMS AND METHODOLOGY — 40%
   A. Data organization
      • Data types
      • Data structures and implementation techniques
   B. Program control and structure
      • Iteration and recursion
      • Procedures, functions, methods and exception handlers
      • Concurrency, communication and synchronization
   C. Programming languages and notation
      • Constructs for data organization and program control
      • Scope, binding and parameter passing
      • Expression evaluation

D. Software engineering
- Formal specifications and assertions
- Verification techniques
- Software development models, patterns and tools

E. Systems
- Compilers, interpreters and run-time systems
- Operating systems, including resource management and protection/security
- Networking, Internet and distributed systems
- Databases
- System analysis and development tools

II. COMPUTER ORGANIZATION AND ARCHITECTURE — 15%
A. Digital logic design
- Implementation of combinational and sequential circuits
- Optimization and analysis

B. Processors and control units
- Instruction sets
- Computer arithmetic and number representation
- Register and ALU organization
- Data paths and control sequencing

C. Memories and their hierarchies
- Performance, implementation and management
- Cache, main and secondary storage
- Virtual memory, paging and segmentation

D. Networking and communications
- Interconnect structures (e.g., buses, switches, routers)
- I/O systems and protocols
- Synchronization

E. High-performance architectures
- Pipelining superscalar and out-of-order execution processors
- Parallel and distributed architectures

III. THEORY AND MATHEMATICAL BACKGROUND — 40%
A. Algorithms and complexity
- Exact and asymptotic analysis of specific algorithms
- Algorithmic design techniques (e.g., greedy, dynamic programming, divide and conquer)
- Upper and lower bounds on the complexity of specific problems
- Computational complexity, including NP-completeness

B. Automata and language theory
- Models of computation (finite automata, Turing machines)
- Formal languages and grammars (regular and context-free)
- Decidability

C. Discrete structures
- Mathematical logic
- Elementary combinatorics and graph theory
- 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 a mathematical background in the areas of calculus and linear algebra as applied to computer science.

Preparing for a Subject Test

GRE Subject Test questions are designed to measure skills and knowledge gained over a long period of time. Although you might increase your scores to some extent through preparation a few weeks or months before you take the test, last minute cramming is unlikely to be of further help. The following information may be helpful.

- A general review of your college courses is probably the best preparation for the test. However, the test covers a broad range of subject matter, and no one is expected to be familiar with the content of every question.

- Use this practice book to become familiar with the types of questions in the GRE Computer Science Test, taking note of the directions. If you understand the directions before you take the test, you will have more time during the test to focus on the questions themselves.

Test-Taking Strategies

The questions in the practice test in this book illustrate the types of multiple-choice questions in the test. When you take the actual test, you will mark your answers on a separate machine-scorable answer sheet. Total testing time is two hours and fifty minutes; there are no separately timed sections. Following are some general test-taking strategies you may want to consider.

- Read the test directions carefully, and work as rapidly as you can without being careless. For each question, choose the best answer from the available options.

- All questions are of equal value; do not waste time pondering individual questions you find extremely difficult or unfamiliar.

- You may want to work through the test quite rapidly, first answering only the questions about which you feel confident, then going back and answering questions that require more thought, and concluding with the most difficult questions if there is time.

- If you decide to change an answer, make sure you completely erase it and fill in the oval corresponding to your desired answer.

- Questions for which you mark no answer or more than one answer are not counted in scoring.

- Your score will be determined by subtracting one-fourth the number of incorrect answers from the number of correct answers. If you have some knowledge of a question and are able to rule out one or more of the answer choices as incorrect, your chances of selecting the correct answer are improved, and answering such questions will likely improve your score. It is unlikely that pure guessing will raise your score; it may lower your score.

- Record all answers on your answer sheet. Answers recorded in your test book will not be counted.

- Do not wait until the last five minutes of a testing session to record answers on your answer sheet.

COMPUTER SCIENCE TEST
PRACTICE BOOK

What Your Scores Mean

Your raw score — that is, the number of questions you answered correctly minus one-fourth of the number you answered incorrectly — is converted to the scaled score that is reported. This conversion ensures that a scaled score reported for any edition of a Subject Test is comparable to the same scaled score earned on any other edition of the same test. Thus, equal scaled scores on a particular Subject Test indicate essentially equal levels of performance regardless of the test edition taken. Test scores should be compared only with other scores on the same Subject Test. (For example, a 680 on the Computer Science Test is not equivalent to a 680 on the Mathematics Test.)

Before taking the test, you may find it useful to know approximately what raw scores would be required to obtain a certain scaled score. Several factors influence the conversion of your raw score to your scaled score, such as the difficulty of the test edition and the number of test questions included in the computation of your raw score. Based on recent editions of the Computer Science Test, the following table gives the range of raw scores associated with selected scaled scores for three different test editions. (Note that when the number of scored questions for a given test is greater than the number of actual scaled score points, it is likely that two or more raw scores will convert to the same scaled score.) The three test editions in the table that follows were selected to reflect varying degrees of difficulty. Examinees should note that future test editions may be somewhat more or less difficult than the test editions illustrated in the table.

Range of Raw Scores* Needed to Earn Selected Scaled Scores on Three Computer Science Test Editions That Differ in Difficulty

Scaled Score	Raw Scores		
	Form A	Form B	Form C
800	44	40-41	38-39
700	30	25	23
600	16-17	13	10
500	3	2	0
Number of Questions Used to Compute Raw Score			
	69	70	67

*Raw Score = Number of Correct Answers minus one-fourth the number of incorrect answers, rounded to the nearest integer.

For a particular test edition, there are many ways to earn the same raw score. For example, on the edition listed above as "Form A," a raw score of 30 would earn a scaled score of 700. Below are a few of the possible ways in which a scaled score of 700 could be earned on the edition

Examples of Ways to Earn a Scaled Score of 700 on the Edition Labeled as "Form A"

Raw Score	Questions Answered Correctly	Questions Answered Incorrectly	Questions Not Answered	Number of Questions Used to Compute Raw Score
30	30	0	39	69
30	34	15	20	69
30	37	30	2	69

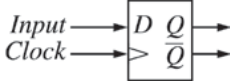# COMPUTER SCIENCE TEST
Time—170 minutes
70 Questions

The pages at the beginning of the Computer Science Test containing the "Notation, Conventions, and Definitions" were revised in 2010. Test editions with a copyright date of 2010 or later will contain the following revised pages. Note that revisions were made to # 4, 5, 7, 8, 9, 10, 11, and 13.

Notation, Conventions, and Definitions:

In this test a reading knowledge of modern programming languages is assumed. The following notational conventions and definitions are used.

1. All numbers are assumed to be written in decimal notation unless otherwise indicated.

2. $\lfloor x \rfloor$ denotes the greatest integer that is less than or equal to x.

3. $\lceil x \rceil$ denotes the least integer that is greater than or equal to x.

4. $g(n) = O(f(n))$ means there exist $C > 0$ and $N > 0$ such that $|g(n)| \leq Cf(n)$ for all $n > N$.

   $g(n) = \Omega(f(n))$ means there exist $C > 0$ and $N > 0$ such that $g(n) \geq Cf(n)$ for all $n > N$.

   $g(n) = \Theta(f(n))$ means $g(n) = O(f(n))$ and $g(n) = \Omega(f(n))$.

   $g(n) = o(f(n))$ means $g(n)/f(n) \to 0$ as $n \to \infty$.

   $g(n) = \omega(f(n))$ means $g(n)/f(n) \to \infty$ as $n \to \infty$.

5. $\exists$ denotes "there exists".

   $\forall$ denotes "for all".

   $\to$ denotes "implies"; $\supset$ is also used to denote "implies"; and $\Rightarrow$ is also used to denote "implies".

   $\neg$ denotes "not"; "$\overline{A}$" is also used to denote "not A".

   $\vee$ denotes "inclusive or"; $+$ also denotes "inclusive or", e.g., $P + Q$ can denote "P or Q".

   $\oplus$ denotes "exclusive or".

   $\wedge$ denotes "and"; also, juxtaposition of statements can denote "and", e.g., PQ can denote "P and Q".

   A Boolean formula is <u>satisfiable</u> if it is true under some assignment of Boolean values for its variables.

   A Boolean formula is a <u>tautology</u> (or is <u>valid</u>) if it is true under all assignments of Boolean values for its variables.

6. $\varnothing$ denotes the empty set.

   If A and B denote sets, then:

       $A \cup B$ is the set of all elements that are in A or in B or in both;

       $A \cap B$ is the set of all elements that are in both A and B;

       $A - B$ is the set of all elements in A that are not in B;

       $\overline{A}$ is the set of all elements not in A that are in some specified universal set;

       $|A|$ is the cardinality of A.

7. In a string expression, ϵ denotes the empty string.

   A string also denotes the singleton set containing just that string.

   If S and T denote sets of strings, then:

       ST denotes the concatenation of S and T;

       $S + T$ denotes $S \cup T$;

       S^n denotes $\underbrace{SS...S}_{n \text{ factors}}$;

       S^* denotes $\epsilon + S + S^2 + S^3 + ...$;

       S^+ denotes $S + S^2 + S^3 + ...$.

8. The complement of a language L over an alphabet Σ is defined to be $\Sigma^* - L$.

9. Unless otherwise indicated in a question, an undirected graph $G = (V, E)$ has no self-loops or multiedges (multiple edges between the same pair of vertices). In addition, assume that $|V| > 0$.

10. In a logic diagram:

    represents an AND gate.

    represents an inclusive OR gate.

    represents an exclusive OR gate.

    represents a NOT gate.

    represents a NAND gate.

    represents a NOR gate.

11. *Input* → D Q / *Clock* → $\bar{Q}$ represents a D-type flip-flop, which stores the value of its D input when clocked.

12. In a finite automaton diagram, states are represented by circles, with final (or accepting) states indicated by two concentric circles. The start state is labeled "Start". An arc from state s to state t labeled a indicates a transition from s to t on input a. A label a/b indicates that this transition produces an output b. A label $a_1, a_2, ..., a_k$ indicates that the transition is made on any of the inputs $a_1, a_2, ..., a_k$.

13. Unless otherwise stated, concurrent programs execute on a platform in which shared memory is not cached, assignment statements execute atomically, and the scheduling algorithm is fair, but no other assumptions are made. The following mechanisms are used to control concurrency:

    `fork()` creates a new child process with memory and register contents initially identical to the parent process. The operation returns `0` in the child process and `1` in the parent process.

    `join()` blocks until all child processes have terminated.

For a semaphore `s`:

    the operation `P(s)` blocks until `s > 0` and then decrements `s` before continuing.

    the operation `V(s)` increments `s` and wakes up one process blocked on `s` (if any).

A binary semaphore is a semaphore restricted to have a value of either `0` or `1`, with an initial value of `1`.

A mutual exclusion lock `m`, with accompanying `lock(m)` and `unlock(m)` operations, is synonymous with a binary semaphore.

14. A <u>loop invariant</u> for a `while` statement

        while B do S

is an assertion that is true each time the guard B is evaluated during execution of the `while` statement.

Practice Test

To become familiar with how the administration will be conducted at the test center, first remove the answer sheet (pages 55 and 56). Then go to the back cover of the test book (page 50) and follow the instructions for completing the identification areas of the answer sheet. When you are ready to begin the test, note the time and begin marking your answers on the answer sheet.

GRE

29

GRADUATE RECORD EXAMINATIONS®

COMPUTER SCIENCE TEST

*Do not break the seal
until you are told to do so.*

*The contents of this test are confidential.
Disclosure or reproduction of any portion
of it is prohibited.*

THIS TEST BOOK MUST NOT BE TAKEN FROM THE ROOM.

COMPUTER SCIENCE TEST
Time—170 minutes
70 Questions

Notation, Conventions, and Definitions:

In this test a reading knowledge of modern programming languages is assumed. The following notational conventions and definitions are used.

1. All numbers are assumed to be written in decimal notation unless otherwise indicated.

2. $\lfloor x \rfloor$ denotes the greatest integer that is less than or equal to x.

3. $\lceil x \rceil$ denotes the least integer that is greater than or equal to x.

4. $g(n) = O(f(n))$ denotes "$g(n)$ has order at most $f(n)$" and means that there exist positive constants C and N such that $|g(n)| \le Cf(n)$ for all $n > N$.

    $g(n) = \Omega(f(n))$ denotes "$g(n)$ has order at least $f(n)$" and for this test means that there exist positive constants C and N such that $g(n) \ge Cf(n)$ for all $n > N$.

    $g(n) = \Theta(f(n))$ denotes "$g(n)$ has the same order as $f(n)$" and means that there exist positive constants C_1, C_2, and N such that $C_1 f(n) \le g(n) \le C_2 f(n)$ for all $n > N$.

5. $\exists$ denotes "there exists".

    $\forall$ denotes "for all".

    $\rightarrow$ denotes "implies"; $\supset$ is also used to denote "implies"; and $\Rightarrow$ is also used to denote "implies".

    $\neg$ denotes "not"; "$\overline{A}$" is also used as meaning "$\neg A$".

    $\vee$ denotes "inclusive or"; $+$ also denotes "inclusive or", e.g., $P + Q$ can denote "P or Q".

    $\oplus$ denotes "exclusive or".

    $\wedge$ denotes "and"; also, juxtaposition of statements can denote "and", e.g., PQ can denote "P and Q".

    A Boolean formula is <u>satisfiable</u> if it is true under some assignment of Boolean values for its variables.

    A Boolean formula is a <u>tautology</u> (or is <u>valid</u>) if it is true under all assignments of Boolean values for its variables.

6. $\varnothing$ denotes the empty set.

    If A and B denote sets, then:

    $A \cup B$ is the set of all elements that are in A or in B or in both;

    $A \cap B$ is the set of all elements that are in both A and B;

    $A - B$ is the set of all elements in A that are not in B;

    $\overline{A}$ is the set of all elements not in A that are in some specified universal set;

    $|A|$ is the cardinality of A.

7. In a string expression, if S and T denote strings or sets of strings, then:

    An empty string is denoted by ϵ or by Λ;

    ST denotes the concatenation of S and T;

    $S + T$ denotes $S \cup T$ or $\{S, T\}$, depending on context;

    S^n denotes $\underbrace{SS...S}_{n \text{ factors}}$;

    S^* denotes $\epsilon + S + S^2 + S^3 + ...$;

    S^+ denotes $S + S^2 + S^3 + ...$

8. In a grammar:

$\alpha \rightarrow \beta$ represents a production in the grammar.

$\alpha \Rightarrow \beta$ means β can be derived from α by the application of exactly one production.

$\alpha \overset{*}{\Rightarrow} \beta$ means β can be derived from α by the application of zero or more productions.

Unless otherwise specified,

   (i)   symbols appearing on the left-hand side of productions are nonterminal symbols; the remaining symbols are terminal symbols;
   (ii) the leftmost symbol of the first production is the start symbol; and
   (iii) the start symbol is permitted to appear on the right-hand side of productions.

9. In a logic diagram:
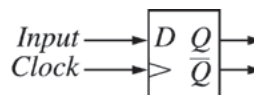
 represents an AND gate.

 represents an inclusive OR gate.

 represents an exclusive OR gate.

 represents a NOT gate.

 represents a NAND gate.

 represents a NOR gate.

10. represents a D-type flip-flop, which stores the value of its D input when clocked.

11. Binary tree traversal is defined recursively as follows.

   Preorder: visit the root, traverse the left subtree, traverse the right subtree
   Inorder: traverse the left subtree, visit the root, traverse the right subtree
   Postorder: traverse the left subtree, traverse the right subtree, visit the root

12. In a finite automaton diagram, states are represented by circles, with final (or accepting) states indicated by two concentric circles. The start state is labeled "Start". An arc from state s to state t labeled a indicates a transition from s to t on input a. A label a/b indicates that this transition produces an output b. A label $a_1, a_2, ..., a_k$ indicates that the transition is made on any of the inputs $a_1, a_2, ..., a_k$.

13. For a program segment S and predicates P and Q, to say that the triple $\{P\}S\{Q\}$ is <u>partially correct</u> means that if P is true before initiation of S, then Q is true upon termination of S. To say that $\{P\}S\{Q\}$ is <u>totally correct</u> means that it is partially correct and S terminates for all inputs for which P is true.

Given that $\{P\}S\{Q\}$ is partially correct, a <u>precondition</u> is any assertion that implies P, and a <u>postcondition</u> is any assertion that is implied by Q.

14. A <u>loop invariant</u> for a `while` statement

$$\text{while } B \text{ do } S$$

is an assertion that is true each time the guard B is evaluated during execution of the `while` statement.

Directions: Each of the questions or incomplete statements below is followed by five suggested answers or completions. Select the one that is the best of the choices offered and then mark the corresponding space on the answer sheet.

1. Consider the following pseudocode of the recursive function `goAgain`.

```
// assume N is a positive integer
integer goAgain ( integer a, integer N )
begin
    if ( N == 1 ) then
        return a
    else
        return a + goAgain ( a, N/2 )
end
```

What is the order of growth, as a function of N, of the number of times the function `goAgain` is recursively executed?

(A) $\Theta(\log N)$ (B) $\Theta(N)$ (C) $\Theta(N \log N)$ (D) $\Theta(N^2)$ (E) $\Theta(N^3)$

---

2. Consider the binary heap shown below that uses an array `a[] = [9, 7, 4, 2, 6, 1, 3]` to store its elements.
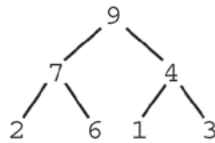
What will be the values of the remaining elements in the array after one delete-max operation from the heap?

(A) [7, 4, 2, 6, 1, 3]

(B) [7, 4, 6, 2, 1, 3]

(C) [7, 4, 6, 2, 3, 1]

(D) [7, 6, 4, 2, 1, 3]

(E) [7, 6, 4, 2, 3, 1]

GO ON TO THE NEXT PAGE.

3. Assume that any n-bit positive integer x is stored as a linked list of bits so that the first element of the list is the least significant bit. For example, $x = 14 = 1110_2$ is stored as the linked list $(0, 1, 1, 1)$ of size $n = 4$.

For this data structure, the operation that replaces x by $\left\lfloor \dfrac{x}{8} \right\rfloor$ can be done in

(A) $\Theta(1)$ steps

(B) $\Theta(\log n)$ steps

(C) $\Theta(n)$ steps

(D) $\Theta(n \log n)$ steps

(E) $\Theta(n^2)$ steps

GO ON TO THE NEXT PAGE.

Questions 4-5 refer to the following information.

Consider the following nondeterministic finite state automaton over alphabet $\{x, y\}$ with start state S.
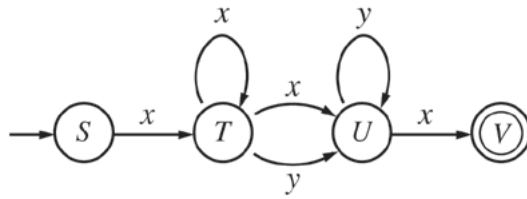
4. Which of the following is the regular expression corresponding to the automaton above?

   (A) $xxx + yyx$

   (B) x^3y^2x

   (C) x^*y^*x

   (D) $xx^*(x + y)y^*x$

   (E) x^*xy^*x

5. Which of the following grammars over alphabet $\{x, y\}$ generates the language recognized by the automaton above?

   (A) $S \rightarrow xT$
   $T \rightarrow xT \mid xU \mid yU$
   $U \rightarrow yU \mid xV$

   (B) $S \rightarrow xT$
   $T \rightarrow xT \mid xU \mid yU$
   $U \rightarrow yU \mid x$

   (C) $S \rightarrow xT \mid T$
   $T \rightarrow xT \mid xU \mid yU \mid T \mid U$
   $U \rightarrow yU \mid xV \mid V \mid x$

   (D) $S \rightarrow xV$
   $T \rightarrow xT \mid yU$
   $U \rightarrow yU \mid xV$

   (E) $S \rightarrow xT$
   $T \rightarrow xT \mid T$
   $U \rightarrow yU \mid V$
   $V \rightarrow xV \mid x$

GO ON TO THE NEXT PAGE.

6. Consider the following function.

```
int f ()
{
    int k, result;
    result = 0;
    for ( k = 0; k < 5; k++ )
    {
        if ( ( k % 3 ) == 1 )
            result = result + k;
        else
            result = result + 1;
    }
    return result;
}
```

What value is returned as a result of the call `f ()` ?

(A) 5   (B) 6   (C) 7   (D) 8   (E) 9

---

7. A privileged instruction may be executed only while the hardware is in kernel mode. Which of the following is LEAST likely to be a privileged instruction?

(A) An instruction that changes the value of the program counter

(B) An instruction that sends output to a printer

(C) An instruction that modifies a memory management register

(D) An instruction that halts the CPU

(E) An instruction that resets the computer's time-of-day clock

---

GO ON TO THE NEXT PAGE.

8. Which data structure would be most appropriate to implement a collection of values with the following three characteristics?

- Items are retrieved and removed from the collection in FIFO order.

- There is no a priori limit on the number of items in the collection.

- The size of an item is large relative to the storage required for a memory address.

(A) Singly-linked list, with head and tail pointers

(B) Doubly-linked list, with only a head pointer

(C) Array

(D) Binary tree

(E) Hash table

9. A researcher is preparing a questionnaire with 6 questions. The only possible responses to each question are "Yes," "Maybe," and "No." The researcher wants to know how many people answer with any given combination of responses. A programmer is designing a data structure to collate the responses to this questionnaire. The programmer decides to use a base structure containing 6 memory locations, one for each question. Each element will contain a 2 for "Yes," a 1 for "Maybe," and a 0 for "No." One person's response may look like this:

1	2	2	1	0	1

The programmer then declares an array with one slot for each possible combination of responses. What will be the number of entries in the array?

(A) 3   (B) 6   (C) 216   (D) 324   (E) 729

10. Consider the following truth table and implementation.

Possibly Faulty Implementation

Truth Table

P	Q	R	X	Y
0	0	0	1	0
0	0	1	1	0
0	1	0	0	1
0	1	1	0	0
1	0	0	0	0
1	0	1	0	0
1	1	0	0	1
1	1	1	0	0
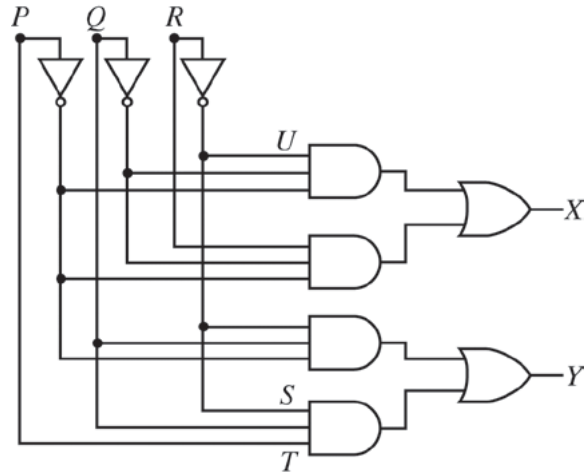
Which of the following statements is (are) true about the circuit shown?

 I. The line labeled S is correct.

 II. The line labeled T is correct.

 III. The line labeled U is correct.

(A) None (B) I only (C) II only (D) III only (E) I, II, and III

11. Suppose problem A is **NP**-complete and problem B is in **NP** but is not necessarily **NP**-complete. Which of the following statements is (are) necessarily true?

    I. A polynomial-time algorithm for A implies $\mathbf{P} = \mathbf{NP}$.

    II. A polynomial-time algorithm for B implies $\mathbf{P} = \mathbf{NP}$.

    III. A polynomial-time algorithm for A implies a polynomial-time algorithm for B.

(A) I only   (B) II only   (C) I and II only   (D) I and III only   (E) I, II, and III

---

12. Two classical algorithms for finding a minimum spanning tree in a graph are Kruskal's algorithm and Prim's algorithm. Which of the following are the design paradigms used by these algorithms?

	Kruskal's algorithm	Prim's algorithm
(A)	The greedy method	The greedy method
(B)	The greedy method	Dynamic programming
(C)	Dynamic programming	The greedy method
(D)	Dynamic programming	Divide and conquer
(E)	Divide and conquer	Dynamic programming

GO ON TO THE NEXT PAGE.

13. Consider the following pseudocode.

```
program main ()
begin
    integer a, b, c, i
    a = 6
    b = 7
    c = 8
    i = fun(a, b, c)
    print i, a, b, c
end

integer fun (integer x, integer y, integer z)
begin
    if (x > 6) then
        y = 25
    z = x + y
    return y + z
end
```

If fun uses call-by-reference, what values would be printed as a result of executing the pseudocode?

	i	a	b	c
(A)	6	20	8	7
(B)	6	20	13	7
(C)	20	6	7	8
(D)	20	6	7	13
(E)	56	6	25	31

GO ON TO THE NEXT PAGE.

14. Which of the following is the most appropriate data structure to store the symbol table of a compiler?

(A) Linked list　　(B) Hash table　　(C) Priority queue　　(D) Queue　　(E) Set

15. A software requirements specification is

　I. a contract between developers and clients specifying what the developers will produce for the clients.

　II. a specification of the features that the target software deliverable must have.

　III. a specification of the personnel and resources that will be committed to the software development effort.

(A) I only　　(B) II only　　(C) I and II only　　(D) I and III only　　(E) I, II, and III

16. A full binary tree is a rooted tree in which every internal node has exactly two children. How many internal nodes are there in a full binary tree with 500 leaves?

(A) 250　　(B) 499　　(C) 500　　(D) 501　　(E) 1,000

GO ON TO THE NEXT PAGE.

17. Consider the following instruction sequence in a single-issue in-order 5-stage pipeline (IF, ID, EX, MEM, and WB).

Opcode	Destination	Source1	Source2	
ADD	R1	R2	R3	← First instruction to enter the pipeline
SUB	R2	R3	R1	
MUL	R1	R2	R3	← Last instruction to enter the pipeline

How many data hazards does the ID stage need to detect for this instruction sequence?

(A) 1   (B) 2   (C) 3   (D) 4   (E) 5

---

18. Suppose that Professor X develops a new model of computation, called a neutron machine. Which of the following would be a consequence of the Church-Turing thesis?

(A) No neutron machine can solve the Traveling Salesman problem in polynomial time.

(B) No neutron machine can solve the Maximum Matching problem for bipartite graphs in polynomial time.

(C) No neutron machine can determine whether the decimal expansion of π has 7 consecutive 7s.

(D) No neutron machine can simulate a given Turing machine in polynomial time.

(E) No neutron machine can determine in polynomial time whether a given Turing machine halts when its input tape is initially blank.

GO ON TO THE NEXT PAGE.

19. Consider the following pseudocode in which all variables are integers and $m \geq 1$.

Pseudocode I

```
sum = 0
for i = 1 to m
    sum = sum + i
output ( sum )
```

The following pseudocode is intended to produce the same output as Pseudocode I.

Pseudocode II

```
sum = 0
i = <initial value>
while ( <condition> )
{
    i = i + 1
    sum = sum + i
}
output ( sum )
```

Which of the following can be used to replace *<initial value>* and *<condition>* so that Pseudocode II will produce the same output as Pseudocode I ?

	<initial value>	*<condition>*
(A)	0	i < m - 1
(B)	0	i < m
(C)	0	i < m + 1
(D)	1	i < m
(E)	1	i < m + 1

GO ON TO THE NEXT PAGE.

20. Consider the following page reference string: 1, 2, 3, 4, 2, 1, 5, 6, 2, 1, 2, 3, 7, 6, 3, 2, 1

Assuming there are 4 page frames available and that all frames are initially empty, what is the total number of page faults that would occur for the page reference string above if the least-recently-used (LRU) replacement policy is used?

(A) 6   (B) 7   (C) 9   (D) 10   (E) 11

---

21. Consider the following binary search tree.
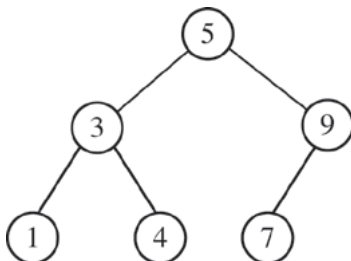
Starting from an empty binary search tree, the insertion of which of the following sequences of integer keys could produce the binary tree above?

(A) 5, 9, 1, 7, 3, 4

(B) 5, 7, 4, 9, 3, 1

(C) 5, 4, 7, 3, 9, 1

(D) 5, 3, 4, 9, 1, 7

(E) 5, 3, 1, 7, 9, 4

GO ON TO THE NEXT PAGE.

22. Consider a relational database schema with the following instance of a relation $R(A, B, C)$.

A	B	C
1	1	1
1	2	3
2	2	4

Assuming the instance above is valid, which of the following functional dependencies must be FALSE?

I. $A \rightarrow B$

II. $A \rightarrow C$

III. $AB \rightarrow C$

(A) I only   (B) II only   (C) I and II   (D) I and III   (E) II and III

---

23. A computer memory system has a 64KB byte-addressable main memory with 16-bit addresses. This same system has a one-level cache memory that can hold 16 blocks of data, where each block contains 16 bytes. Assuming this is a direct-mapped cache, to which cache block will main memory address $9A81_{16}$ map?

(A) 0_{16} (B) 1_{16} (C) 8_{16} (D) 9_{16} (E) A_{16}

---

24. Let $M_0, M_1, M_2, \ldots,$ be an effective enumeration of all Turing machines. Which of the following problems is (are) decidable?

I. Given a natural number n, does M_n starting with an empty tape halt in fewer than n steps?

II. Given a natural number n, does M_n starting with an empty tape halt in exactly n steps?

III. Given a natural number n, does M_n starting with an empty tape halt after at least n steps?

(A) I only   (B) II only   (C) III only   (D) I and II   (E) I and III

25. Consider the pseudocode block below. Assume that the variables `x` and `y` are integers.

```
// Precondition: ????
x ← x + y
y ← x - y
x ← x - y
// Postcondition: (( x = 3 ) and ( y = 8 ))
```

Which of the following would be an appropriate precondition to guarantee the stated postcondition?

(A) (x = 3) and (y = 8)

(B) (x = 8) and (y = 3)

(C) (x = 8) and (y = 11)

(D) (x = 11) and (y = 3)

(E) (x = 11) and (y = 8)

---

26. Which of the following best describes the difference between paging and segmentation?

(A) Paging breaks a process' virtual memory into physical units of the same size, whereas segmentation breaks a process' virtual memory into logical units that are typically of different sizes.

(B) Paging suffers from external fragmentation, whereas segmentation suffers from internal fragmentation.

(C) Paging requires page tables for address translation, whereas segmentation does not require segment tables for address translation.

(D) Paging requires one page table per process, whereas segmentation requires only one global segment table for the entire system.

(E) Page tables are typically very small, whereas segment tables are always very large.

GO ON TO THE NEXT PAGE.

27. Consider the following two fragments of Java programs.

```java
public void P1 (int x, int y, int z)
{
    if ((x != 0) && ((y / x) == z))
        z = z + 1;
    System.out.println("x = " + x + " y = " + y + " z = " + z);
}


public void P2 (int x, int y, int z)
{
    if (((y / x) == z) && (x != 0))
        z = z + 1;
    System.out.println("x = " + x + " y = " + y + " z = " + z);
}
```

Which of the following is true?

(A) For all x, y, and z, P1(x, y, z) and P2(x, y, z) have the same behavior.

(B) For all x and y, there exists z such that P1(x, y, z) and P2(x, y, z) behave differently.

(C) For all x and z, there exists y such that P1(x, y, z) and P2(x, y, z) behave differently.

(D) For all y and z, there exists x such that P1(x, y, z) and P2(x, y, z) behave differently.

(E) For all x, y, and z, P1(x, y, z) and P2(x, y, z) behave differently.

---

28. Which of the following statements about positive integers is NOT true?

(A) If x is a composite integer, then x has a prime divisor less than or equal to the square root of x.

(B) There are infinitely many prime integers.

(C) Integers a and b are congruent modulo m if and only if there is an integer k such that $a = b + km$.

(D) If a divides bc, then either a divides b or a divides c.

(E) If a divides b and b divides c, then a divides c.

GO ON TO THE NEXT PAGE.

29. Five wireless nodes—A, B, C, D, and E—are arranged in a line. Any pair of adjacent nodes are within range of each other, and nonadjacent nodes do not interfere. All nodes execute the Carrier Sense Multiple Access/ Collision Avoidance (CSMA/CA) protocol with Ready to Send/Clear to Send (RTS/CTS) exchange prior to transmission. If the link bandwidth is L bits per second for successful transmissions and reliable transmission is desired, what is the maximum possible total bandwidth attainable for this network, in bits per second?

(A) L (B) $2L$ (C) $3L$ (D) $5L$ (E) $6L$

---

30. Consider the following directed graph.
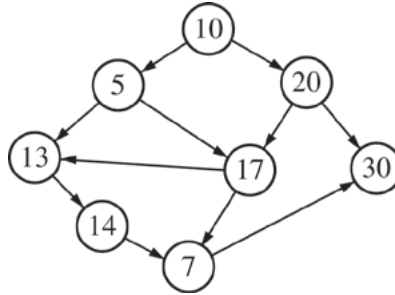
Which of the following is a topological sort of the nodes of the graph?

(A)   5, 7, 10, 13, 14, 17, 20, 30

(B) 10, 5, 13, 14, 7, 30, 17, 20

(C) 10, 5, 13, 17, 20, 14, 7, 30

(D) 10, 5, 20, 13, 17, 30, 14, 7

(E) 10, 20, 5, 17, 13, 14, 7, 30

GO ON TO THE NEXT PAGE.

31. Consider a brute-force password-guessing attack that can submit authentication requests at a rate of one every millisecond. Assume that a password consists of 1–6 characters from a 10-symbol alphabet. In the <u>average</u> case, approximately how many seconds would it take to determine the password using this type of attack?

(A) 100   (B) 500   (C) 555   (D) 1,000   (E) 1,111

---

32. Consider the following two problems.

Nearest Neighbors: Given an unsorted array of n floating-point numbers as input, return two of the numbers that are closest in value to each other.

Farthest Neighbors: Given an unsorted array of n floating-point numbers as input, return two of the numbers that are farthest in value from each other.

Assume that the only operations allowed on the data are

- comparing the values of two entries in the array and identifying the larger value;

- comparing the distance between two array entries (the absolute value of the difference between the two array entries) with the distance between two other array entries;

- swapping two entries in the array.

Further assume that each allowed operation has unit cost. What are the worst-case optimal asymptotic running times for algorithms that solve the two problems?

	Nearest Neighbors	Farthest Neighbors
(A)	$\Theta(n \log n)$	$\Theta(n)$
(B)	$\Theta(n \log n)$	$\Theta(n \log n)$
(C)	$\Theta(n^2)$	$\Theta(n)$
(D)	$\Theta(n^2)$	$\Theta(n \log n)$
(E)	$\Theta(n^2)$	$\Theta(n^2)$

GO ON TO THE NEXT PAGE.

33. Which of the following typically occurs when a procedure call is executed on a processor?

    I. Program counter is updated.

    II. Stack pointer is updated.

    III. Data cache is flushed to avoid aliasing problems.

(A) I only   (B) II only   (C) I and II only   (D) I and III only   (E) I, II, and III

---

34. The routing table below uses the longest prefix match in its routing decisions.

Network Address/Length	Next Hop
196.94.8.0/21	S
196.94.16.0/24	T
196.94.0.0/16	U

What are the next hops for each of the following two destination addresses?

	Next Hop to 196.94.10.7	Next Hop to 196.94.34.9
(A)	S	T
(B)	S	U
(C)	T	T
(D)	T	U
(E)	U	U

---

35. Consider a recursive algorithm for sorting an array of $n \geq 2$ integers that works as follows.

    (a) If there are only 2 elements to be sorted, compare them and swap them if they are out of order.

    (b) Otherwise, do the following steps in order.

        (1) Recursively sort the first $n - 1$ elements of the array.
        (2) In the resulting array, recursively sort the last $n - 1$ elements.
        (3) In the resulting array, recursively sort the first 2 elements of the array.

What is the asymptotic running time complexity of this algorithm measured in terms of the number of comparisons made?

(A) $\Theta(n \log n)$ (B) $\Theta(n^2)$ (C) $\Theta(n^3)$ (D) $\Theta(2^n)$ (E) $\Theta(3^n)$

GO ON TO THE NEXT PAGE.

36. Consider the following Java code.

```java
class BaseClass
{
    int data = 101;
    public void print ()
    {
        System.out.print ( data + " " );
    }
    public void fun ()
    {
        print ();
    }
}

class SubClass extends BaseClass
{
    int data = 202;
    public void print ()
    {
        System.out.print ( data + " " );
    }
}

class TestClass
{
    public static void main ( String[] args )
    {
        BaseClass obj = new SubClass ();
        obj.print ();
        obj.fun ();
        System.out.print ( obj.data );
    }
}
```

What is printed by the execution of `TestClass.main ()` ?

(A) `101 101 101`

(B) `101 202 202`

(C) `202 101 101`

(D) `202 202 101`

(E) `202 202 202`

GO ON TO THE NEXT PAGE.

37. A hash function h maps 16-bit inputs to 8-bit hash values. What is the largest k such that in any set of 1,000 inputs, there are at least k inputs that h maps to the same hash value?

(A) 3   (B) 4   (C) 10   (D) 64   (E) 256

---

38. Consider three processes P1, P2, and P3 with respective arrival times of 0 ms, 10 ms, and 20 ms and respective processing times of 30 ms, 15 ms, and 30 ms. The three processes are preemptively scheduled on a single-CPU system using the shortest-remaining-processing-time-first scheduling policy. Which of the following shows the order in which the processes complete, from first to last?

(A)   P1   P2   P3

(B)   P1   P3   P2

(C)   P2   P1   P3

(D)   P2   P3   P1

(E)   P3   P1   P2

---

39. Which of the following statements about fixed-length and variable-length instruction set architectures (ISAs) is (are) true?

   I. Variable-length ISAs allow for a smaller code size over fixed-length ISAs.

  II. Fixed-length ISAs simplify instruction fetch and decode over variable-length ISAs.

 III. Variable-length ISAs require more registers than fixed-length ISAs.

(A) I only   (B) II only   (C) I and II only   (D) II and III only   (E) I, II, and III

GO ON TO THE NEXT PAGE.

40. Consider the following pseudocode program.

```
int i

main ()
{
    i = 3
    S ()
    R ()
}

void S ()
{
    print i     // prints the value of i on the current line of output
    print " "   // prints a blank space on the current line of output
}

void R ()
{
    int i
    i = 2
    S ()
}
```

What is the output of the program if the pseudocode uses either static (lexical) scoping or dynamic scoping?

	Static Scoping	Dynamic Scoping
(A)	3 2	3 2
(B)	3 3	2 2
(C)	3 3	2 3
(D)	3 3	3 2
(E)	3 3	3 3

GO ON TO THE NEXT PAGE.

41. Given a directed graph $G = (V, E)$, it is convenient to represent the connectivity properties of G using an associated directed acyclic graph $G' = (V', E')$, where the vertices in V' are the strongly connected components of G and for $S, T \in V'$, (S, T) is in E' if and only if there exist $u \in S$ and $v \in T$ such that $(u, v) \in E$.

Let G be the graph shown below.
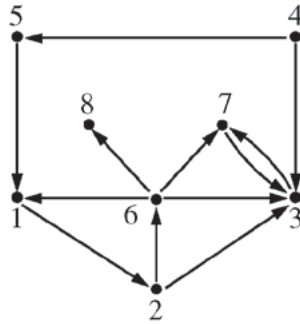
Which of the following is its associated directed acyclic graph G'?

(A)

(B)

(C)

(D) $\{1, 2, 3, 4, 5, 6, 7\}$ ────────▶ $\{8\}$

(E)

GO ON TO THE NEXT PAGE.

42. An algorithm's real-time readiness (RTR) ratio is defined as the ratio of its average-case running time to its worst-case running time.

Which of the following algorithms has an RTR ratio closest to 0 ?

(A) Bubblesort   (B) Heapsort   (C) Insertion sort   (D) Mergesort   (E) Quicksort

---

43. Which of the following properties must be true of a Minimum Spanning Tree (MST) of a connected graph G with at least 3 edges?

    I. The MST must contain the shortest edge of G.

    II. The MST must contain the second-shortest edge of G.

    III. The MST can never contain the longest edge of G.

(A) None   (B) I only   (C) I and II only   (D) I and III only   (E) I, II, and III

---

44. A color (RGB) raster-scan graphics system provides 18 bits per pixel and uses no color lookup table. If black and white count as shades of gray, how many different shades of gray does the system offer?

(A) 64   (B) 255   (C) 256   (D) 262,143   (E) 262,144

45. If the delay through a single-bit adder is 3 (measured in gate delays) to the sum output and 2 to the carry output, what is the delay through a k-bit ripple-carry adder?

(A) $2k - 1$ (B) $2k + 1$ (C) $3k - 1$ (D) $3k$ (E) $3k + 1$

---

46. Consider the following pseudocode fragment.

```
i = 0;
j = 0;
k = 0;

while (i < 10 and j < 10)
{
    if (A[i] > B[j])
    {
        C[k] = A[i];
        k = k + 1  ;
        i = i + 1  ;
    }
    else
    {
        C[k] = B[j];
        k = k + 1  ;
        j = j + 1  ;
    }
}
```

Which of the following could serve as a loop invariant for the `while` loop above?

I. `i < 10 or j < 10`

II. `i < 11 and j < 11`

III. `k = i + j`

(A) I only   (B) I and II only   (C) I and III only   (D) II and III only   (E) I, II, and III

GO ON TO THE NEXT PAGE.

47. Consider the following concurrent tasks, in which each assignment statement executes atomically. Initially, the shared variables `x` and `y` are set to `0`.

Task 1
```
x = 1
a = y
```

Task 2
```
y = 1
b = x
```

At the end of the concurrent tasks, the values of `a` and `b` are examined. Which of the following must be true?

I. (a == 0) → (b == 1)

II. (b == 0) → (a == 1)

III. (a == 1) → (b == 1)

(A) I only   (B) II only   (C) III only   (D) I and II only   (E) I, II, and III

---

48. Five balls are placed randomly into four boxes, labeled A, B, C, and D. Each ball has an equal chance of being placed into any box. What is the expected total number of balls in boxes A and B ?

(A) 1   (B) 1.25   (C) 1.5   (D) 2   (E) 2.5

---

49. One garbage collection algorithm is semispace copying collection. Which of the following characteristics of garbage collection apply to semispace copying collection?

I. Collects dead objects that reference each other

II. Incurs overhead on every assignment operation to a reference variable

III. Avoids fragmentation

(A) None   (B) I and II only   (C) I and III only   (D) II and III only   (E) I, II, and III

50. Which of the following statements about caches is (are) true?

    I. A direct-mapped cache can have a lower miss rate than an associative cache of the same size (number of blocks).

    II. Programs with high spatial locality have a low cache miss rate primarily because the exact same addresses are re-referenced.

    III. Programs with high temporal locality have a low cache miss rate primarily because the exact same addresses are re-referenced.

(A) I only   (B) II only   (C) III only   (D) I and II   (E) I and III

---

51. A *k-sorted array* is a nearly sorted array in which no element is more than k locations away from its final position in the sorted array. Thus, a 0-sorted array is completely sorted and every array of size n is n-sorted.

Suppose that A is a k-sorted array of size n. If insertion sort is used to sort A, what is the order of growth of the number of comparisons performed by the sorting algorithm in the worst case?

(A) $\Theta(k)$ (B) $\Theta(kn)$ (C) $\Theta(k^2 n)$ (D) $\Theta(n \log_k n)$ (E) $\Theta(n^2)$

---

52. The subtype principle describes when one type may be substituted for another. Which of the following is true?

(A) An instance of a <u>subtype</u> may be used in any expression in which an instance of the <u>supertype</u> may be used because the <u>subtype</u> must support a <u>superset</u> of the operations supported by the <u>supertype</u>.

(B) An instance of a <u>subtype</u> may be used in any expression in which an instance of the <u>supertype</u> may be used because the <u>subtype</u> must support a <u>subset</u> of the operations supported by the <u>supertype</u>.

(C) An instance of a <u>supertype</u> may be used in any expression in which an instance of the <u>subtype</u> may be used because the <u>subtype</u> must support a <u>superset</u> of the operations supported by the <u>supertype</u>.

(D) An instance of a <u>supertype</u> may be used in any expression in which an instance of the <u>subtype</u> may be used because the <u>subtype</u> must support a <u>subset</u> of the operations supported by the <u>supertype</u>.

(E) Instances of <u>subtypes</u> and <u>supertypes</u> may be used interchangeably.

GO ON TO THE NEXT PAGE.

53. Consider a regular language L over $\{0, 1\}$. Which of the following languages over $\{0, 1\}$ must also be regular?

    I. $\{w \in L \mid$ the length of w is even$\}$

    II. $\{w \in L \mid$ the length of w is prime$\}$

    III. $\{w \in L \mid$ the length of w is an integer power of 2$\}$

(A) None   (B) I only   (C) III only   (D) I and III only   (E) I, II, and III

---

54. In order to create a good solution for the mutual exclusion problem for concurrent processes, which of the following conditions must hold?

    I. No process should have to wait forever to enter its critical region.

    II. No process running outside of its critical region may block other processes from entering their critical region.

    III. There should be no assumptions about the speed or number of CPUs.

(A) None   (B) I and II only   (C) I and III only   (D) II and III only   (E) I, II, and III

---

55. What is the negation of the predicate $\forall x \, \exists y \left(p(y) \to q(x) \right)$?

(A) $\forall x \, \exists y \left(p(y) \to \neg \, q(x) \right)$

(B) $\forall x \, \exists y \left(p(y) \wedge \neg \, q(x) \right)$

(C) $\exists x \, \forall y \left(p(y) \to \neg \, q(x) \right)$

(D) $\exists x \, \forall y \left(q(x) \to \neg \, p(y) \right)$

(E) $\exists x \, \forall y \left(p(y) \wedge \neg \, q(x) \right)$

56. Consider a single-issue processor with an in-order five-stage pipeline (IF, ID, EX, MEM, and WB) and with the following characteristics.

- The register file can be written and then read in the same cycle.

- The pipeline does not have other bypassing/forwarding hardware.

- Instructions, including loads and stores, spend only one cycle in the MEM stage.

Consider the following MIPS-like instructions in which the destination register is the first (leftmost) register. The instruction set supports (i) register-indirect addressing, which is indicated by using parentheses and (ii) displacement addressing, which is indicated as an integer offset from a register indirect value.

```
loadword $t0, ($t1)
add $t2, $t2, $t0
loadword $t0, 4($t1)
add $t1, $t1, 1
add $t2, $t2, $t0
sub $t3, $t1, $t0
```

How many stall cycles do the instructions incur?

(A) 0   (B) 1   (C) 2   (D) 3   (E) 4

GO ON TO THE NEXT PAGE.

57. A hiker faces the 0/1 Knapsack problem. There are 7 items to be packed into the knapsack, each with value v_i and weight w_i as shown in the following table.

i	1	2	3	4	5	6	7
v_i	3	6	8	1	2	5	7
w_i	7	3	5	1	4	2	6

The knapsack, which is initially empty, can hold a maximum weight of 24, so some item(s) must be left behind, and fractions of items cannot be packed. The optimality criterion is to maximize the total value of the items that are placed in the knapsack. The hiker fills the knapsack one item at a time, using a heuristic algorithm that is greedy on value density, where the value density of an item is its value/weight ratio. When this heuristic algorithm is used, what is the total value of the items that are packed, and is this total optimal?

	Total Value	Optimal/Not Optimal
(A)	29	Optimal
(B)	29	Not optimal
(C)	30	Optimal
(D)	30	Not optimal
(E)	32	Optimal

58. Amdahl's Law pertains to the speedup achieved when running a program on parallel processors versus using a single serial processor. In this context, speedup is the ratio of original running time to improved running time. According to Amdahl's Law, approximately how much speedup could we expect for an unlimited number of processors if 10 percent of a program is sequential and the remaining part is ideally parallel?

(A) 10X   (B) 20X   (C) 40X   (D) 90X   (E) Infinite

59. Let $T(n)$ be defined by $T(0) = T(1) = 4$ and $T(n) = T\left(\left\lfloor \dfrac{n}{2} \right\rfloor\right) + T\left(\left\lfloor \dfrac{n}{4} \right\rfloor\right) + cn$ for all integers $n \geq 2$, where c is a positive constant. What is the asymptotic growth of $T(n)$?

(A) $\Theta(\log n)$ (B) $\Theta(n)$ (C) $\Theta(n \log n)$ (D) $\Theta(n^2)$ (E) $\Theta(n^{\log_3 4})$

---

60. Suppose that stacks and queues are provided as opaque data types, offering only operations to add elements, to remove elements, and to test for emptiness. Suppose that a programmer wants to count the number of elements in a given stack or queue C, which is currently in some state t, using only one auxiliary stack or queue D. The structures C and D can be used in <u>any</u> way possible based on the methods they offer, but C must be restored to its state t after counting its elements.

Counting elements as described above is possible for which of the following data types?

    I. C is a queue and D is a queue.

    II. C is a stack and D is a stack.

    III. C is a queue and D is a stack.

(A) None   (B) I and II only   (C) I and III only   (D) II and III only   (E) I, II, and III

---

61. Which of the following problems is (are) known to be solvable in running time $O(n^3)$?

    I. Find the shortest path from a given start vertex to a given end vertex in a directed graph on n vertices with nonnegative integer weights.

    II. Find the longest simple path from a given start vertex to a given end vertex in a directed graph on n vertices with nonnegative integer weights.

    III. Find the longest path from a given start vertex to a given end vertex in a directed acyclic graph (DAG) on n vertices with nonnegative integer weights.

(A) I only   (B) I and II only   (C) I and III only   (D) II and III only   (E) I, II, and III

GO ON TO THE NEXT PAGE.

62. The Fibonacci sequence F_n is defined by $F_1 = 1$, $F_2 = 1$, and $F_n = F_{n-2} + F_{n-1}$ for all integers $n \geq 3$. What is the minimal number of D flip-flops required (along with combinational logic) to design a counter circuit that outputs the first seven Fibonacci numbers (i.e., F_1 through F_7) and then wraps around?

(A) 3　　　(B) 4　　　(C) 5　　　(D) 6　　　(E) 7

---

63. An algorithm takes a list of 2^n numbers $\left[a_1, a_2, ..., a_{2^n}\right]$ and replaces it with $\left[b_1, b_2, ..., b_{2^{n-1}}\right]$, where $b_1 = \max\{a_1, a_2\}$, $b_2 = \max\{a_3, a_4\}$, and so on. Then it performs the same operation on the resulting list (replacing each pair of consecutive elements with their maximum), and it continues doing the same until there are only two elements left in the list. For instance, if the initial list is [3, 7, 6, 8, 2, 1, 4, 5], then after the first run, it becomes [7, 8, 2, 5] and then [8, 5].

Suppose that the elements of the initial list are the integers 1 through 64 in random order. What is the probability that the number 63 will appear in the final two-element list?

(A) $\dfrac{1}{63}$　　　(B) $\dfrac{1}{4}$　　　(C) $\dfrac{32}{63}$　　　(D) $\dfrac{62}{63}$　　　(E) 1

GO ON TO THE NEXT PAGE.

64. Consider the following instruction sequence for a hypothetical RISC processor.

$$\text{T.} \quad R1 \leftarrow R2 + R3$$
$$\text{U.} \quad R4 \leftarrow R5 + R6$$
$$\text{V.} \quad R5 \leftarrow R7 + R8$$
$$\text{W.} \quad R9 \leftarrow R5 + R1$$
$$\text{X.} \quad R10 \leftarrow R4 + R1$$
$$\text{Y.} \quad R11 \leftarrow R10 + R1$$
$$\text{Z.} \quad R9 \leftarrow R1 + R4$$

Which of the following is a possible legal execution order for the instructions on an out-of-order processor without register renaming?

(A) T, U, X, V, W, Z, Y

(B) T, U, X, V, Z, W, Y

(C) T, V, U, X, W, Y, Z

(D) U, T, V, Y, X, W, Z

(E) U, T, X, W, Y, Z, V

65. Consider the following three statements.

    I. If language L is recognized by a nondeterministic finite automaton, then L is recognized by a deterministic finite automaton.

    II. If language L is recognized by a nondeterministic pushdown automaton, then L is recognized by a deterministic pushdown automaton.

    III. If language L is recognized by a nondeterministic polynomial-time Turing machine, then L is recognized by a deterministic polynomial-time Turing machine.

Which of the following best describes what is currently known about the statements?

(A) One of the three statements is known to be false; it is not currently known whether the other two statements are true or false.

(B) Two of the three statements are known to be false; it is not currently known whether the other statement is true or false.

(C) One of the three statements is known to be true; another statement is known to be false; it is not currently known whether the remaining statement is true or false.

(D) One of the three statements is known to be true; the other two statements are known to be false.

(E) Two of the three statements are known to be true; the other statement is known to be false.

---

66. Query optimizers typically use summaries of data distributions to estimate the sizes of the intermediate tables generated during query processing. One popular such summarization scheme is a *histogram*, whereby the input range is partitioned into buckets and a cumulative count is maintained of the number of tuples falling in each bucket. The distribution within a bucket is assumed to be uniform for the purposes of estimation.

The following shows one such histogram for a relation R on a discrete attribute a with domain [1..10].

    • **Bucket 1:** Range = [1..2], Cumulative tuple count = 6

    • **Bucket 2:** Range = [3..8], Cumulative tuple count = 30

    • **Bucket 3:** Range = [9..10], Cumulative tuple count = 10

What is the estimated size of the self-join operation $R \bowtie R$?

(A) 46   (B) 218   (C) 248   (D) 1,036   (E) 5,672

GO ON TO THE NEXT PAGE.

67. Recall that a predicate logic statement is contingent if its truth value depends on the choice of the universe and on the interpretations of the predicate symbol S and the constant symbol b involved.

Consider the following predicate logic statements in which b, x, and y are elements of the universe U.

I. $\forall x\,(S(x, b) \rightarrow \exists y\, S(x, y))$

II. $\forall x\, \exists y\, S(x, y) \rightarrow \exists y\, \forall x\, S(x, y)$

III. $\forall x\,(\neg S(x, x) \leftrightarrow S(b, x))$

Which of the following best describes the predicate logic statements?

	Statement I	Statement II	Statement III
(A)	Always true	Always false	Contingent
(B)	Always true	Contingent	Always false
(C)	Always true	Contingent	Contingent
(D)	Contingent	Always true	Always false
(E)	Contingent	Always true	Always true

---

68. In the graph below, each edge label represents the probability that the connection between its endpoints is working. If these probabilities are mutually independent, what is the probability that there is a path of working edges from S to T?
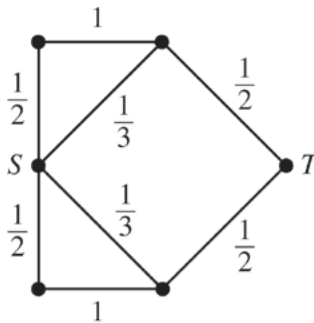
(A) $\dfrac{1}{3}$ (B) $\dfrac{5}{12}$ (C) $\dfrac{5}{9}$ (D) $\dfrac{2}{3}$ (E) $\dfrac{5}{6}$

GO ON TO THE NEXT PAGE.

69. Suppose that in RSA encryption, the public encryption key is the pair $(e, n) = (3, 55)$ and the private decryption key is the pair $(d, n) = (d, 55)$, where $d < n$ is a positive integer. What is the value of d ?

(A) 13   (B) 27   (C) 37   (D) 39   (E) 54

---

70. If L_1 is a decidable language and L_2 is an undecidable language, then $L_1 \cup L_2$, the union of L_1 and L_2, is

(A) possibly finite and possibly infinite, but definitely decidable

(B) possibly finite and possibly infinite, but definitely undecidable

(C) infinite, but possibly decidable and possibly undecidable

(D) infinite and decidable

(E) infinite and undecidable

STOP
If you finish before time is called, you may check your work on this test.

NO TEST MATERIAL ON THIS PAGE

I

SUBJECT TEST

A. Print and sign your full name in this box:

PRINT: _____
(LAST)   (FIRST)   (MIDDLE)

SIGN: _____

Copy this code in box 6 on your answer sheet. Then fill in the corresponding ovals exactly as shown.

6. TITLE CODE
2 9 7 8 5

Copy the Test Name and Form Code in box 7 on your answer sheet.

TEST NAME   **Computer Science**

FORM CODE   **GR0629**

GRADUATE RECORD EXAMINATIONS SUBJECT TEST

B. The Subject Tests are intended to measure your achievement in a specialized field of study. Most of the questions are concerned with subject matter that is probably familiar to you, but some of the questions may refer to areas that you have not studied.

Your score will be determined by subtracting one-fourth the number of incorrect answers from the number of correct answers. Questions for which you mark no answer or more than one answer are not counted in scoring. If you have some knowledge of a question and are able to rule out one or more of the answer choices as incorrect, your chances of selecting the correct answer are improved, and answering such questions will likely improve your score. It is unlikely that pure guessing will raise your score; it may lower your score.

You are advised to use your time effectively and to work as rapidly as you can without losing accuracy. Do not spend too much time on questions that are too difficult for you. Go on to the other questions and come back to the difficult ones later if you can.

YOU MUST INDICATE ALL YOUR ANSWERS ON THE SEPARATE ANSWER SHEET. No credit will be given for anything written in this examination book, but you may write in the book as much as you wish to work out your answers. After you have decided on your response to a question, fill in the corresponding oval on the answer sheet. BE SURE THAT EACH MARK IS DARK AND COMPLETELY FILLS THE OVAL. Mark <u>only one</u> answer to each question. No credit will be given for multiple answers. Erase all stray marks. If you change an answer, be sure that all previous marks are erased completely. Incomplete erasures may be read as intended answers. Do not be concerned that the answer sheet provides spaces for more answers than there are questions in the test.

Example:

What city is the capital of France?

(A) Rome
(B) Paris
(C) London
(D) Cairo
(E) Oslo

Sample Answer

CORRECT ANSWER PROPERLY MARKED

IMPROPER MARKS

DO NOT OPEN YOUR TEST BOOK UNTIL YOU ARE TOLD TO DO SO.

Educational Testing Service

Educational Testing Service
Princeton, New Jersey 08541

Scoring Your Subject Test

The Computer Science Test scores are reported on a 200 to 990 score scale in ten-point increments. The actual range of scores is smaller, and it varies from edition to edition because different editions are not of precisely the same difficulty. However, this variation in score range is usually small and should be taken into account mainly when comparing two very high scores. In general, differences between scores at the 99th percentile should be ignored. The score conversion table on page 53 shows the score range for this edition of the test only.

The worksheet on page 52 lists the correct answers to the questions. Columns are provided for you to mark whether you chose the correct (C) answer or an incorrect (I) answer to each question. Draw a line across any question you omitted, because it is not counted in the scoring. At the bottom of the page, enter the total number correct and the total number incorrect. Divide the total incorrect by 4 and subtract the resulting number from the total correct. Then round the result to the nearest whole number. This will give you your raw total score. Use the total score conversion table to find the scaled total score that corresponds to your raw total score.

Example: Suppose you chose the correct answers to 38 questions and incorrect answers to 22. Dividing 22 by 4 yields 5.5. Subtracting 5.5 from 38 equals 32.5, which is rounded to 33. The raw score of 33 corresponds to a scaled score of 740.

Worksheet for the Computer Science Test, Form GR0629 Only
Answer Key and Percentages* of Examinees Answering
Each Question Correctly

QUESTION Number	Answer	P +	TOTAL C	I	QUESTION Number	Answer	P +	TOTAL C	I
1	A	80			36	D	6		
2	E	28			37	B	33		
3	A	60			38	C	89		
4	D	83			39	C	57		
5	B	73			40	D	82		
6	D	79			41	A	63		
7	A	26			42	E	57		
8	A	82			43	C	34		
9	E	84			44	A	35		
10	E	70			45	B	42		
11	D	55			46	E	35		
12	A	35			47	D	60		
13	D	90			48	E	69		
14	B	84			49	C	15		
15	C	55			50	E	23		
16	B	73			51	B	49		
17	B	45			52	A	62		
18	E	46			53	B	48		
19	B	58			54	E	52		
20	D	42			55	E	50		
21	D	90			56	D	12		
22	C	43			57	B	54		
23	C	27			58	A	58		
24	D	62			59	B	14		
25	B	94			60	E	28		
26	A	51			61	C	33		
27	D	65			62	A	15		
28	D	50			63	C	59		
29	B	25			64	A	46		
30	E	45			65	C	34		
31	C	40			66	B	5		
32	A	19			67	B	30		
33	C	75			68	C	16		
34	B	32			69	B	8		
35	D	33			70	C	26		

Correct (**C**) _____

Incorrect (**I**) _____

Total Score:

C – **I**/4 = _____

Scaled Score (SS) = _____

* The P+ column indicates the percent of COMPUTER SCIENCE Test examinees who answered each
question correctly; it is based on a sample of examinees selected to represent all COMPUTER SCIENCE
Test examinees tested between July 1, 2007, and June 30, 2010.

COMPUTER SCIENCE TEST
PRACTICE BOOK

Score Conversions and Percents Below*
for GRE Computer Science Test,
Form GR0629 Only

TOTAL SCORE					
Raw Score	Scaled Score	%	Raw Score	Scaled Score	%
63-70	890	99	26	690	42
58-62	880	99	24-25	680	38
55-57	870	98	23	670	36
53-54	860	96	22	660	33
50-52	850	94	21	650	30
48-49	840	92	19-20	640	26
46-47	830	89	18	630	23
45	820	86	17	620	20
43-44	810	83	16	610	19
41-42	800	81	14-15	600	15
40	790	77	13	590	13
38-39	780	74	12	580	11
37	770	71	11	570	9
35-36	760	67	9-10	560	7
34	750	63	8	550	6
32-33	740	59	7	540	5
31	730	56	6	530	4
30	720	52	4-5	520	3
28-29	710	49	3	510	2
27	700	46	2	500	2
			0-1	490	1

*Percentage scoring below the scaled score is based on the performance
 of 3,364 examinees who took the COMPUTER SCIENCE Test between July 1,
 2007, and June 30, 2010.

Evaluating Your Performance

Now that you have scored your test, you may wish to compare your performance with the performance of others who took this test. Both the worksheet on page 52 and the table on page 53 use performance data from GRE Computer Science Test examinees.

The data in the worksheet on page 52 are based on the performance of a sample of examinees selected to represent all GRE Computer Science Test examinees tested between July 2007 and June 2010. The numbers in the column labeled "P+" on the worksheet indicate the percentages of examinees in this sample who answered each question correctly. You may use these numbers as a guide for evaluating your performance on each test question.

The table on page 53 contains, for each scaled score, the percentage of examinees tested between July 2007 and June 2010 who received lower scores. Interpretive data based on the scores earned by examinees tested in this three-year period will be used by admissions officers in the 2011-12 testing year. These percentages appear in the score conversion table in a column to the right of the scaled scores. For example, in the percentage column opposite the scaled score of 730 is the number 56. This means that 56 percent of the GRE Computer Science Test examinees tested between July 2007 and June 2010 scored lower than 730. To compare yourself with this population, look at the percentage next to the scaled score you earned on the practice test.

It is important to realize that the conditions under which you tested yourself were not exactly the same as those you will encounter at a test center. It is impossible to predict how different test-taking conditions will affect test performance, and this is only one factor that may account for differences between your practice test scores and your actual test scores. By comparing your performance on this practice test with the performance of other GRE Computer Science Test examinees, however, you will be able to determine your strengths and weaknesses and can then plan a program of study to prepare yourself for taking the GRE Computer Science Test under standard conditions.

DO NOT USE INK

Use only a pencil with soft, black lead (No. 2 or HB) to complete this answer sheet.
Be sure to fill in completely the space that corresponds to your answer choice.
Completely erase any errors or stray marks.

1. NAME
Enter your last name, first name initial (given name), and middle initial if you have one.
Omit spaces, apostrophes, Jr., II., etc.

Last Name only (Family Name or Surname) - First 15 Letters

First Name Initial
Middle Initial

2.
YOUR NAME:
(Print)
Last Name (Family or Surname)   First Name (Given)   M.I.

MAILING ADDRESS:
(Print)
P.O. Box or Street Address

City   State or Province

Country   Zip or Postal Code

CENTER:
Country   City   State or Province

Center Number   Room Number

SIGNATURE:

Q3117-06.07/1   00101-02954 • TF-7TE70   MH/Ivan07190
1   2   3   4

742862

3. DATE OF BIRTH
Month   Day   Year
Jan.   Feb.   Mar.   April   May   June   July   Aug.   Sept.   Oct.   Nov.   Dec.

4. SOCIAL SECURITY NUMBER
(U.S.A. only)

5. REGISTRATION NUMBER
(from your admission ticket)

6. TITLE CODE
(on back cover of your test book)

7. TEST NAME (on back cover of your test book)

FORM CODE (on back cover of your test book)

8. TEST BOOK SERIAL NUMBER
(number in upper right corner of front cover of your test book)

SHADED AREA FOR ETS USE ONLY

BE SURE EACH MARK IS DARK AND COMPLETELY FILLS THE INTENDED SPACE AS ILLUSTRATED HERE: ●
YOU MAY FIND MORE RESPONSE SPACES THAN YOU NEED. IF SO, PLEASE LEAVE THEM BLANK.

1 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	39 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	77 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
2 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	40 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	78 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
3 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	41 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	79 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
4 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	42 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	80 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
5 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	43 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	81 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
6 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	44 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	82 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
7 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	45 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	83 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
8 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	46 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	84 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
9 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	47 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	85 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
10 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	48 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	86 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
11 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	49 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	87 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
12 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	50 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	88 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
13 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	51 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	89 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
14 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	52 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	90 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
15 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	53 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	91 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
16 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	54 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	92 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
17 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	55 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	93 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
18 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	56 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	94 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
19 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	57 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	95 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
20 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	58 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	96 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
21 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	59 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	97 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
22 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	60 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	98 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
23 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	61 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	99 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
24 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	62 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	100 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
25 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	63 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	101 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
26 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	64 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	102 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
27 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	65 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	103 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
28 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	66 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	104 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
29 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	67 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	105 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
30 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	68 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	106 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
31 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	69 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	107 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
32 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	70 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	108 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
33 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	71 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	109 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
34 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	72 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	110 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
35 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	73 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	111 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
36 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	74 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	112 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
37 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	75 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	113 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
38 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	76 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	114 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ

55

Item responses continued on reverse side.

SUBJECT TEST

**COMPLETE THE
CERTIFICATION STATEMENT,
THEN TURN ANSWER SHEET
OVER TO SIDE 1.**

CERTIFICATION STATEMENT
Please write the following statement below, DO NOT PRINT.
"I certify that I am the person whose name appears on this answer sheet. I also agree not to disclose the contents of the test I am taking today to anyone."
Sign and date where indicated.

SIGNATURE: _____   DATE: ___/___/___
                                                Month Day Year

BE SURE EACH MARK IS DARK AND COMPLETELY FILLS THE INTENDED SPACE AS ILLUSTRATED HERE: ●
YOU MAY FIND MORE RESPONSE SPACES THAN YOU NEED. IF SO, PLEASE LEAVE THEM BLANK.

115 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	147 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	179 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	211 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
116 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	148 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	180 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	212 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
117 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	149 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	181 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	213 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
118 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	150 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	182 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	214 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
119 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	151 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	183 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	215 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
120 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	152 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	184 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	216 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
121 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	153 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	185 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	217 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
122 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	154 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	186 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	218 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
123 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	155 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	187 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	219 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
124 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	156 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	188 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	220 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
125 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	157 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	189 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	221 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
126 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	158 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	190 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	222 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
127 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	159 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	191 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	223 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
128 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	160 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	192 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	224 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
129 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	161 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	193 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	225 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
130 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	162 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	194 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	226 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
131 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	163 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	195 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	227 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
132 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	164 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	196 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	228 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
133 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	165 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	197 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	229 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
134 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	166 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	198 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	230 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
135 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	167 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	199 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	231 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
136 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	168 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	200 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	232 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
137 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	169 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	201 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	233 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
138 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	170 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	202 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	234 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
139 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	171 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	203 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	235 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
140 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	172 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	204 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	236 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
141 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	173 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	205 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	237 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
142 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	174 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	206 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	238 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
143 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	175 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	207 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	239 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
144 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	176 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	208 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	240 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
145 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	177 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	209 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	241 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ
146 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	178 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	210 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ	242 Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ

IF YOU DO NOT WANT THIS ANSWER SHEET TO BE SCORED

If you want to cancel your scores from this test administration, complete A and B below. You will not receive scores for this test. No record of this test or the cancellation will be sent to the recipients you indicated, and there will be no scores for this test on your GRE file.

To cancel your scores from this test administration, you must:

A. Fill in both ovals here . . . ◯ — ◯   B. sign your full name here: _____

TR	TW	TFS	TCS	1R	1W	1FS	1CS	2R	2W	2FS	2CS
				3R	3W	3FS	3CS	4R	4W	4FS	4CS
	FOR ETS USE ONLY										
				5R	5W	5FS	5CS	6R	6W	6FS	6CS