PRACTICING TO TAKE THE GRE®

COMPUTER SCIENCE TEST

AN OFFICIAL FULL-LENGTH EDITION OF THE GRE COMPUTER SCIENCE TEST ADMINISTERED IN 1985-86

GRADUATE RECORD EXAMINATIONS

PRACTICING TO TAKE THE GRE®

COMPUTER SCIENCE TEST

AN OFFICIAL FULL-LENGTH EDITION OF THE GRE COMPUTER SCIENCE TEST ADMINISTERED IN 1985-86

Published by Educational Testing Service
for the Graduate Record Examinations Board

The Graduate Record Examinations Program offers a General Test measuring verbal, quantitative, and analytical abilities and Subject Tests measuring achievement in the following 15 fields:

Biology	Education	Literature in English	Political Science
Chemistry	Engineering	Mathematics	Psychology
Computer Science	Geology	Music	Sociology
Economics	History	Physics	

The tests are administered by Educational Testing Service under policies determined by the Graduate Record Examinations Board, an independent board affiliated with the Association of Graduate Schools and the Council of Graduate Schools.

The Graduate Record Examinations Board has officially made available for purchase one full-length edition of each Subject Test. Two practice books, each containing three General Tests, are also available. These practice books may be purchased by using the order form on page 55.

Individual booklets describing each test and including sample questions are available free of charge for all 15 Subject Tests. These booklets may be requested by writing to:

Graduate Record Examinations
Educational Testing Service
P.O. Box 6014
Princeton, NJ 08541-6014

In association with Warner Books, Inc., a Warner Communications Company.

USA: 0-446-38455-0
CAN: 0-446-38456-9

GRADUATE RECORD EXAMINATIONS

PRACTICING TO TAKE THE

GRE®

COMPUTER SCIENCE TEST

AN OFFICIAL FULL-LENGTH EDITION OF THE GRE COMPUTER SCIENCE TEST ADMINISTERED IN 1985-86

Published by Educational Testing Service
for the Graduate Record Examinations Board

The Graduate Record Examinations Program offers a General Test measuring verbal, quantitative, and analytical abilities and Subject Tests measuring achievement in the following 15 fields:

Biology	Education	Literature in English	Political Science
Chemistry	Engineering	Mathematics	Psychology
Computer Science	Geology	Music	Sociology
Economics	History	Physics	

The tests are administered by Educational Testing Service under policies determined by the Graduate Record Examinations Board, an independent board affiliated with the Association of Graduate Schools and the Council of Graduate Schools.

The Graduate Record Examinations Board has officially made available for purchase one full-length edition of each Subject Test. Two practice books, each containing three General Tests, are also available. These practice books may be purchased by using the order form on page 55.

Individual booklets describing each test and including sample questions are available free of charge for all 15 Subject Tests. These booklets may be requested by writing to:

Graduate Record Examinations
Educational Testing Service
P.O. Box 6014
Princeton, NJ 08541-6014

In association with Warner Books, Inc., a Warner Communications Company.

USA: 0-446-38455-0
CAN: 0-446-38456-9

Practicing to Take the GRE® Computer Science Test

This practice book contains the full-length GRE® Computer Science Test that was given at GRE test centers in December 1985. It has been published on behalf of the Graduate Record Examinations Board to help potential graduate students prepare to take the test.

The book includes information about the purpose of the GRE Subject Tests, a detailed description of the content specifications for the GRE Computer Science Test, and the procedures for developing the test. This information also appears in the descriptive booklet you will receive when you register to take the test. The practice book contains a complete test book, including the general instructions printed on the back cover and inside back cover. Before you take the test at the test center, you will be given time to read these instructions. They show you how to mark your answer sheet properly and give you advice about guessing.

Try to take this practice test under conditions that simulate those in an actual test administration. Use the answer sheet provided on pages 53 and 54 and mark your answers with a number 2 (soft-lead) pencil, as you will do at the test center. Give yourself 2 hours and 50 minutes in a quiet place and work through the test without interruption, focusing your attention on the questions with the same concentration you would use in taking the test to earn a score. Since you will not be permitted to use them at the test center, do not use dictionaries or other books, compasses, rulers, slide rules, calculators, calculator/watch combinations, or any other aids.

After you complete the test, use the work sheet and conversion table on pages 6 and 7 to score your test. The work sheet also shows the percentage of those who took the test in December 1985 who answered each question correctly so that you can compare your performance on the questions with theirs. Evaluating your performance on the questions should help you determine whether you would benefit by reviewing certain courses before taking the test at the test center.

We believe that if you use this practice book as we have suggested, you will be able to approach the testing experience with increased confidence.

PURPOSE OF THE GRE SUBJECT TESTS

The GRE Subject Tests are designed to help graduate school committees and fellowship sponsors assess the qualifications of applicants in their respective subject fields. The tests also provide students with a means of assessing their own competence.

Scores on the tests are intended to indicate students' mastery of the subject matter emphasized in many undergraduate programs. Since past achievement is usually a good indicator of future performance, the scores aid in predicting students' probable success in advanced study. Because the tests are standardized, the test scores permit comparison of the competence of students from different institutions with different undergraduate programs.

The Graduate Record Examinations Board recommends that scores on the Subject Tests be evaluated 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 transcripts showing the range of courses taken and the grades earned, letters of recommendation, and GRE General Test scores.

DEVELOPMENT OF THE COMPUTER SCIENCE TEST

Each new edition of the Computer Science Test is developed by a committee of examiners composed of specialists in various aspects of the field who come from undergraduate and graduate faculties representative of different types of institutions and different regions of the United States. In selecting members of this committee, the GRE Program staff seeks the advice of the Association for Computing Machinery and the Computer Society of the Institute of Electrical and Electronic Engineers.

Subject-matter and measurement specialists on the ETS staff assist the committee of examiners. They provide information and advice about methods of test construction and help prepare the questions and assemble the test.

Because of the diversity of undergraduate curricula in computer science, it is not possible, within the limitations of a test, to cover all the material that examinees may have studied. The examiners, therefore, try to select questions that sample the basic knowledge and understanding most important for successful graduate study in the field. The committee works to keep the test up-to-date. New editions are developed regularly so the test content changes steadily but gradually, much like most curricula. When a new edition is introduced into the program, it is equated; that is, the scores are related to those on previous editions by statistical methods so that scores from all active editions are directly comparable. Although they do not contain the same questions, all editions of the test are constructed according to equivalent specifications for content and level of difficulty and each measures equivalent knowledge and skills.

After a new edition of the Computer Science Test has been taken by examinees at an international test administration, the performance of the examinees on each question is analyzed. If this analysis and the accompanying appraisal of content reveals that a question is not satisfactory—that it is ambiguous or inappropriate for the group taking the test—the answers to that question are not used in computing the scores.

CONTENT OF THE GRE COMPUTER SCIENCE TEST

The test consists of about 80 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—35%

A. Data organization
   1. Abstract data types (e.g., stacks, queues, lists, strings, trees, sets)
   2. Implementations of data types (e.g., pointers, hashing, encoding, packing, address arithmetic)
   3. File organization (e.g., sequential, indexed, multilevel)
   4. Data models (e.g., hierarchical, relational, network)

B. Organization of program control
   1. Iteration and recursion
   2. Functions, procedures, and exception handlers
   3. Concurrent processes, interprocess communication, and synchronization

C. Programming languages and notation
   1. Applicative *versus* procedural languages
   2. Control and data structure
   3. Scope, extent, and binding
   4. Parameter passing
   5. Expression evaluation

D. Design and development
   1. Program specification
   2. Development methodologies
   3. Development tools

E. Systems
   1. Examples (e.g., compilers, operating systems)
   2. Performance models
   3. Resource management (e.g., scheduling, storage allocation)
   4. Protection and security

II. COMPUTER ORGANIZATION AND ARCHITECTURE—20%

A. Logic design
   1. Implementation of combinational and sequential circuits
   2. Functional properties of digital integrated circuits

B. Processors and control units
   1. Instruction sets, register and ALU organization
   2. Control sequencing, register transfers, microprogramming, pipelining

C. Memories and their hierarchies
   1. Speed, capacity, cost
   2. Cache, main, secondary storage
   3. Virtual memory, paging, segmentation devices

D. I/O devices and interfaces
   1. Functional characterization, data rate, synchronization
   2. Access mechanism, interrupts

E. Interconnection
   1. Bus and switch structures
   2. Network principles and protocols
   3. Distributed resources

III. **THEORY—20%**

A. **Automata and language theory**

1. **Regular languages (e.g., finite automata, nondeterministic finite automata, regular expressions)**
2. **Context-free languages (e.g., notations for grammars, properties such as emptiness, ambiguity)**
3. **Special classes of context-free grammars (e.g., LL, LR, precedence)**
4. **Turing machines and decidability**
5. **Processors for formal languages (e.g., parsers, parser generators)**

B. **Correctness of programs**

1. **Formal specifications and assertions (e.g., pre- and postassertions, loop invariants, invariant relations of a data structure)**
2. Verification techniques (e.g., predicate transformers, Hoare axioms)

C. Analysis of algorithms

1. Exact or asymptotic analysis of the best, worst, or average case of the time and space complexity of specific algorithms
2. Upper and lower bounds on the complexity of specific problems
3. NP-completeness

IV. COMPUTATIONAL MATHEMATICS—20%

A. Discrete structures—Basic elements of:

1. Abstract algebra
2. Mathematical logic, including Boolean algebra
3. Combinatorics
4. Graph theory
5. Set theory
6. Discrete probability
7. Recurrence relations

B. Numerical mathematics

1. Computer arithmetic
2. Classical numerical algorithms
3. Linear algebra

V. SPECIAL TOPICS—5%

Typical topics might include modeling and simulation, information retrieval, artificial intelligence, computer graphics, and data communications.

TEST-TAKING STRATEGY

Presumably, if you are about to take the GRE Computer Science Test, you have completed or nearly completed an undergraduate major in that subject. Reviewing your curriculum is probably the best way for you to prepare to take the test. Because the test provides reliable measurement over a broad range of subject matter, you should not expect to be familiar with the content of every question.

When you take the test, read the test directions carefully and work as rapidly as you can without being careless. Do not spend too much time pondering questions you find extremely difficult or unfamiliar because no question carries greater weight than any other.

You receive one "raw score" point for a right answer and nothing for an omission; one-fourth of a point is lost for each wrong answer. As a result of this procedure, random guessing will probably not increase your score, so it is not a useful strategy. However, if you have some knowledge about a question and can eliminate one or more of the answer choices as wrong, your chance of getting the right answer is improved and, on the average, it will be to your advantage to answer the question. Each raw score is converted to a scaled score for reporting.

WORK SHEET for the COMPUTER SCIENCE Test, Form GR8629
Answer Key and Percentage* of Examinees Answering Each Question Correctly

QUESTION Number	QUESTION Answer	P+	TOTAL R	TOTAL W
1	D	64		
2	C	22		
3	D	83		
4	B	29		
5	A	44		
6	B	6		
7	D	95		
8	C	69		
9	B	47		
10	C	21		
11	C	52		
12	B	10		
13	B	61		
14	B	39		
15	A	58		
16	A	75		
17	B	54		
18	D	49		
19	E	54		
20	E	41		
21	E	22		
22	C	19		
23	D	15		
24	C	44		
25	B	18		
26	B	19		
27	E	46		
28	A	32		
29	C	47		
30	A	69		
31	C	45		
32	A	60		
33	B	57		
34	C	47		
35	B	62		

Right (R) __________

Wrong (W) __________

QUESTION Number	QUESTION Answer	P+	TOTAL R	TOTAL W
36	A	55		
37	D	30		
38	E	29		
39	B	49		
40	D	22		
41	D	9		
42	A	28		
43	C	23		
44	A	46		
45	D	52		
46	B	49		
47	D	10		
48	E	39		
49	A	16		
50	D	40		
51	C	14		
52	B	34		
53	A	16		
54	C	62		
55	D	20		
56	C	83		
57	A	42		
58	E	69		
59	C	21		
60	E	18		
61	D	46		
62	A	18		
63	B	29		
64	D	23		
65	A	31		
66	C	34		
67	C	16		
68	D	67		
69	D	37		
70	E	8		

Right (R) __________

Wrong (W) __________

QUESTION Number	QUESTION Answer	P+	TOTAL R	TOTAL W
71	A	21		
72	B	32		
73	E	12		
74	D	13		
75	D	17		
76	C	17		
77	A	6		
78	B	35		
79	C	11		
80	A	27		

Right (R) __________

Wrong (W) __________

Total Score

R − W/4 = __________

Scaled Score (SS) = __________

*Estimated P+ for the group of examinees who took the GRE Computer Science Test in a recent three year period.

HOW TO SCORE YOUR TEST

The work sheet on page 6 lists the correct answers to the questions. Columns are provided for you to mark whether you chose the right (R) answer or a wrong (W) answer to each question. Draw a line across any question you omitted, because it is not counted in the scoring. At the bottom of each "total" column, enter the number right and the number wrong. Then add the three column totals across to get the total right and total wrong. Divide the total wrong by 4 and subtract the resulting number from the total right. This is the adjustment made for guessing. Then round the result to the nearest whole number. This will give you your raw total score. Use the total score conversion table below to find the scaled total score that corresponds to your raw total score.

Example: Suppose you chose the right answers to 32 questions and wrong answers to 26. Dividing 26 by 4 yields 6.5. Subtracting 6.5 from 32 equals 25.5, which is rounded to 26. The raw score of 26 corresponds to a scaled score of 630.

SCORE CONVERSIONS AND PERCENTS BELOW*
FOR GRE COMPUTER SCIENCE TEST, Form GR8629

TOTAL SCORE					
Raw Score	Scaled Score	%	Raw Score	Scaled Score	%
79-80	990	99	35	690	79
78	980	99	33-34	680	76
76-77	970	99	32	670	73
75	960	99	31	660	69
73-74	950	99	29-30	650	66
72	940	99	28	640	62
70-71	930	99	26-27	630	58
69	920	99	25	620	55
67-68	910	99	23-24	610	51
66	900	99	22	600	47
64-65	890	99	20-21	590	43
63	880	99	19	580	39
61-62	870	99	17-18	570	35
60	860	99	16	560	32
58-59	850	99	14-15	550	28
57	840	99	13	540	25
56	830	99	11-12	530	22
54-55	820	98	10	520	19
53	810	98	9	510	16
51-52	800	97	7-8	500	14
50	790	96	6	490	11
48-49	780	95	4-5	480	9
47	770	94	3	470	8
45-46	760	93	1-2	460	6
44	750	92	0	450	5
42-43	740	90			
41	730	89			
39-40	720	86			
38	710	84			
36-37	700	81			

*Percent scoring below the scaled score based on the performance of the 11,458 examinees who took the GRE Subject Test in Computer Science between October 1, 1981, and September 30, 1984.

EVALUATING YOUR PERFORMANCE

Now that you have scored your test, you may wish to see how your scores compare with those earned by others who took this test. For this purpose, the performance of a sample of the examinees who took the test in December 1985 was analyzed. The sample was selected to represent the total population of GRE examinees tested between October 1981 and September 1984. Interpretive data based on the scores earned by these examinees are to be used by admissions officers in 1986-87. By comparing your performance on this practice test with the performance of the analysis sample, you will be able to determine your strengths and weaknesses and can then plan a program of study to prepare yourself for taking the Computer Science Test under standard conditions.

Two kinds of information are provided. On the work sheet you used to determine your score is a column labeled "P+." The numbers in this column indicate the percent of the examinees in the analysis sample who answered each question correctly. In a test of this kind, a question is considered to be of average difficulty if it is answered correctly by about 60 percent (P+ = 60) of the examinees. Use this as a guide for evaluating your performance on the questions that deal with topics covered in the undergraduate courses you have taken. On these questions, you should do relatively well. There are probably some questions on material you have not encountered in your undergraduate program. You may have omitted these questions or guessed at answers, and your performance on them contributes little to your score.

The other kind of information provided is based on the total scores earned by the analysis sample. It appears in the conversion table for total scores in a column to the right of the scaled scores and shows for each total scaled score the percent of the analysis sample who received lower scores. For example, in the percent column opposite the scaled score 550 is the percent 29. This means that 29 percent of the analysis sample examinees scored lower than 550 on this test. Note the percent paired with the total scaled score you made on the practice test. That number is a reasonable indication of your rank among GRE Computer Science Test examinees if you followed the test-taking suggestions in this practice book.

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 differing test-taking conditions will affect test performance, but this is one factor that may account for differences between your practice test scores and your actual test scores.

ADDITIONAL INFORMATION

If you have any questions about any of the information in this book, please write to:

Graduate Record Examinations Program
CN 6000
Princeton, NJ 08541-6000

Before you start timing yourself on the test that follows, we suggest that you remove the answer sheet (page 53) and turn first to the back cover of the test book (page 52), as you will do at the test center, and follow the instructions for completing the identification areas of the answer sheet. Then read the inside back cover instructions (page 51). When you are ready to begin the test, note the time and start marking your answers to the questions on the answer sheet.

FORM GR8629

29

THE 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.

Princeton, N.J. 08541

COMPUTER SCIENCE TEST

Time—170 minutes

80 Questions

Notation and Conventions:

In this test a reading knowledge of Pascal-like languages is assumed. The following notational conventions 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 $f(n)$" and, for purposes of this test, may be taken to mean that $\lim_{n \to \infty} \left| \frac{g(n)}{f(n)} \right|$ is finite.

5. $\exists$ denotes "there exists."
   $\forall$ denotes "for all."
   $\Rightarrow$ denotes "implies."
   $\neg$ denotes "not"; "$\bar{A}$" is also used as meaning "$\neg A$."
   $\vee$ denotes "inclusive or."
   $\oplus$ denotes "exclusive or."
   $\wedge$ denotes "and"; also, juxtaposition of statements denotes "and," e.g., PQ denotes "P and Q".

6. 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; AB also denotes $A \cap B$;
   $\bar{A}$ is the set of all elements not in A that are in some specified universal set.

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 $\underbrace{SS \ldots S}_{n \text{ factors}}$;
   S^* denotes $\epsilon + S + S^2 + S^3 + \ldots$.

GO ON TO THE NEXT PAGE.

8. In a grammar:

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

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

$\alpha \overset{*}{\Longrightarrow} \beta$ 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:

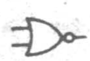
 represents an AND element.

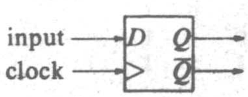
 represents an inclusive OR element.

 represents an exclusive OR element.

 represents a NOT element.

 represents a NAND element.

 represents a NOR element.

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 indicated by the word "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 $a_1, a_2, \ldots, a_k$ indicates that the transition is made on any of the inputs $a_1, a_2, \ldots, a_k$.

GO ON TO THE NEXT PAGE.

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

1. The number of 1's in the binary representation of

$$13*16^3 + 11*16^2 + 9*16 + 3$$

is which of the following?

(A) 7 (B) 8 (C) 9 (D) 10 (E) 12

---

2. Which of the following sorting algorithms has average-case and worst-case running times of $O(n \log n)$?

(A) Bubble sort (B) Insertion sort (C) Merge sort (D) Quicksort (E) Selection sort

---

3. With regard to the Pascal declarations

```
type
  Vector = array[1..10] of integer ;
var
  a   : Vector ;
  b,c : array[1..10] of integer ;
  d   : Vector ;
```

which of the following is FALSE ?

(A) *a* and *b* have structurally equivalent types.
(B) *a* and *d* have name equivalent types.
(C) *b* and *c* have structurally equivalent types.
(D) *b* and *d* have name equivalent types.
(E) *a*, *c*, and *d* have structurally equivalent types.

GO ON TO THE NEXT PAGE.

4. In the NoNicks operating system, the time required by a single file-read operation has four nonoverlapping components:

disk seek time—25 msec
disk latency time—8 msec
disk transfer time—1 msec per 1,000 bytes
operating system overhead—1 msec per 1,000 bytes + 10 msec

In version 1 of the system, the file read retrieved blocks of 1,000 bytes. In version 2, the file read (along with the underlying layout on disk) was modified to retrieve blocks of 4,000 bytes. The ratio of the time required to read a large file under version 2 to the time required to read the same large file under version 1 is approximately

(A) 1:4 (B) 1:3.5 (C) 1:1 (D) 1.1:1 (E) 2.7:1

---

5. Sometimes the object module produced by a compiler includes information (from the symbol table) mapping all source program names to their addresses. The most likely purpose of this information is

(A) for use as input to a debugging aid
(B) to increase the run-time efficiency of the program
(C) for the reduction of the symbol-table space needed by the compiler
(D) to tell the loader where each variable belongs
(E) to tell the operating system what to call the variables

6.

A	B	C	f
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	x
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	x

Consider the logic function in the table above, where x denotes a *don't-care* value. Which of the following statements describes correctly the relation between the minimal sum and the minimal product form of f?

(A) They are logically equivalent by definition.
(B) They are logically equivalent because *don't care*'s are used in the same way.
(C) They are logically equivalent because *don't care*'s do not matter.
(D) They are logically not equivalent by definition.
(E) They are logically not equivalent because *don't care*'s are used in different ways.

GO ON TO THE NEXT PAGE.

Questions 7-8 are based on the following program fragment written in a **Pascal**-like language.

```
L1 : begin
        var a, b, c : integer ;          (1)
        var d, e : real ;
        ______
        ______
        ______
     L2 : begin
             var a, f : integer ;
             var g, h : real ;
             ______
             ______
          end ;
     end
```

Let the designation "block *Li*" refer to all the statements from the **begin** labeled with *Li* to its corresponding **end**.

7. In block *L2* the variables *g* and *h* are best described as

(A) dummy variables (B) parameter variables (C) global variables
(D) local variables (E) recursive variables

8. If the notation *L1-L2* means "the portion of block *L1* that is not in block *L2*," then the scopes of the variables *a* and *b* declared in the statement numbered (1) are

(A) *L1* for *a* and *L1* for *b*
(B) *L1* for *a* and *L1-L2* for *b*
(C) *L1-L2* for *a* and *L1* for *b*
(D) *L1-L2* for *a* and *L1-L2* for *b*
(E) *L2* for *a* and *L2* for *b*

GO ON TO THE NEXT PAGE.