

name _____

Computer Science E-268 Practice Midterm Exam

This exam consists of two parts. Part I has 5 multiple-choice questions worth 3 points each. Part II consists of 3 problems; show all your work on these problems so that partial credit may be awarded if your final answer is wrong but your reasoning is partially correct.

You have 60 minutes to complete the exam. The questions are worth a total of 50 points. In order to properly budget your time, plan on spending **one minute per point**.

You may use any notes, books, or other non-electronic references at your disposal. However, do not waste too much time leafing through your materials. All electronic devices (cell phones, PDAs, etc.) must be turned off for the duration of the exam.

Do all your work on the exam itself. Write clearly. Good luck!

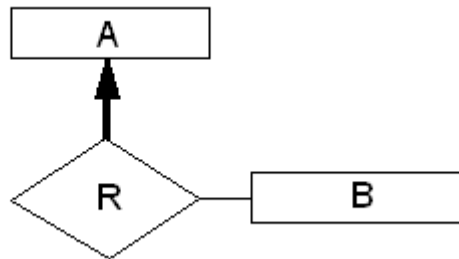
Problem	Max. Score	SCORE
I	15	
II-1	12	
II-2	12	
II-3	11	
TOTAL	50	

Answers to Part I:

question #:	1	2	3	4	5
first choice:					
second choice:					

Part I. Multiple-choice (3 pts. each). Specify up to two answers.

Write your answers in the table at the bottom of the cover sheet. Your first choice for a particular question should be written directly below the question number in the row labeled "first choice." If you decide to provide a second choice for a particular question, write it directly below your first choice in the row labeled "second choice." You will get one point for any problem where your first choice is wrong but your second choice is correct.



1. Given the portion of an ER diagram shown above (with a thickened arrow from R to A), which of the following statements are true?

- I. R connects each entity in A to at least one entity in B
- II. R connects each entity in A to at most one entity in B
- III. R connects each entity in B to at least one entity in A
- IV. R connects each entity in B to at most one entity in A

- A. only I and II
- B. only II and III
- C. only I and IV
- D. only III and IV
- E. all four statements are true

2. Consider a relation R(a, b, c) having the following tuples:

(1, 20, 100), (4, 20, 100), (2, 15, 100), and (1, 18, 100).

Which of the following attributes or attribute combinations could be the primary key of the relation?

- A. attribute a
- B. attribute b
- C. either the combination (a, b) or the combination (a, b, c)
- D. either the combination (b, c) or the combination (a, b, c)
- E. only the combination (a, b)

3. You are given a table named `Al ums` that contains the names and personal information of all graduates of the college that you work for. It includes name and age attributes, and a `state` attribute specifying the state in which a person resides.

Consider the following SQL queries:

- I. `SELECT name, MIN(age)`
 `FROM Al ums`
 `WHERE state = "CA";`
- II. `SELECT name, age`
 `FROM Al ums`
 `WHERE state = "CA"`
 `AND age <= ALL (SELECT age FROM Al ums`
 `WHERE state = "CA");`
- III. `SELECT name, age`
 `FROM Al ums`
 `WHERE state = "CA"`
 `AND age = (SELECT MIN(age) FROM Al ums);`

Which of these queries would successfully find the name and age of the youngest graduate living in California (CA)?

- A. only I
B. only II
C. only III
D. only I and II
E. only II and III
F. I, II, and III
4. You have two relations A and B. A has three tuples, and B has two. What is the maximum number of tuples that could appear in the natural join of A and B?
- A. 2
B. 3
C. 4
D. 5
E. 6

5. The hash table shown at right is being maintained using dynamic linear hashing. The key "cat" was just inserted in the table, and as a result an extra bucket (bucket 4) was added to the table. Which of the keys do we need to consider for possible movement to the new bucket?

- A. only "dog"
- B. only "lion"
- C. only "cat"
- D. both "dog" and "lion"
- E. all four of the keys

000 = 0	dog
001 = 1	frog
010 = 2	lion
011 = 3	cat
100 = 4	

PART II: Answer all three questions in the space provided.

II-1. ER Diagrams (11 points total)

Harvard has offered you free tuition for the spring semester on top of your usual consulting fee if you design a logical schema for a new database. Talking with the people in the administration, you gather the following information:

- The database contains information about professors and courses.
- Professors have a SSN, a name, and a research specialty. Each professor can be uniquely identified by her SSN.
- Each course can be uniquely identified by a course number, and we also want to store the course's name.
- For each course offering, we need to record the professor who teaches the course.
- Each course is taught by exactly one professor.

- Draw a small ER diagram that captures the above information. (6 points)
- Write SQL statements to create the corresponding relations that capture all of the constraints expressed in the ER diagram. If you had trouble completing part a, you can instead write SQL statements to create relations that capture all of the information about the domain given above. (5 points)

II-2. Relational Queries (12 points total)

Consider the following relational schema (keys are underlined):

Product(pid, name, price, manufacturer)

Buys(cid, pid, month, day, year)

Customer(cid, cname, age)

Note that Buys(cid) references Customer(cid), and Buys(pid) references Product(pid).

- a. Write a **single relational algebra query** for the following: "Find the names of all customers who have purchased products that are **not** manufactured by Acme." You may assume that "Acme" appears somewhere in the manufacturer field of all products manufactured by Acme. (6 points)

Do EITHER b or c below, and clearly indicate which one you want us to grade. (6 points)

- b. Write a **single SQL query** for the following: "For each product, determine how many times it was bought in 2010, if at all." The result should be tuples of the form (product name, num times). If a product was not purchased at all in 2010, the second attribute should have a value of 0. For full credit, your answer should *not* use the UNION operator.

- c. Write a **single SQL query** for the following: "Find the names of all customers who have not purchased any Acme products."

II-3. Storage and Indexing (12 points total; 4 points each part)

- a. Assume that a DBMS is using variable-length records that begin with a header of field offsets to store tuples from a slightly simplified Movie relation:

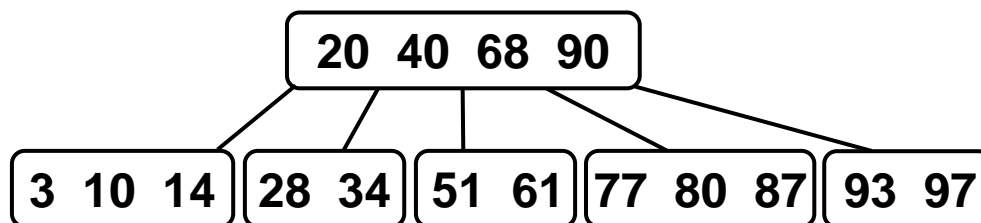
Movie(id CHAR(7), name VARCHAR(64), year INTEGER,
rating VARCHAR(5), runtime INTEGER, earnings_rank INTEGER)

Assuming 1-byte characters and 4-byte integers, what would the record look like for the following tuple?

("1234567", "The King's Speech", 2010, "R", 118, null)

We will accept any reasonable way of representing null, as long as it cannot be confused with a regular, non-null value. Explain your approach to nulls briefly.

- b. Consider the following B-tree of order 2:



What would the tree look like after the following sequence of operations:
insert 65, insert 79, insert 85.

(continued on the next page)

- c. Consider the following hash table, which was constructed using linear hashing with the hash function $h(k) = \text{the length of the string } k$. For example, "True Lies" has a hash code of 9, because it is 9 characters long.

0	"Dinosaur"
1	"True Lies"
2	"Avatar", "Braveheart"
3	"Ray", "Ratatouille", "Michael Clayton"
4	"Philadelphia", "Milk"
5	"Fargo"

Where would "Stuart Little" (length 13) and "Batman Returns" (length 14) be inserted in this table, assuming that they can be added without growing the table?
(You may find it helpful to make use of the chart of decimal-binary equivalents below.)

After these two insertions, if we then increase the size of the table by one bucket, which keys would need to be rehashed, and which of them (if any) would actually be moved to a different bucket?

rehashed:

moved:

decimal-binary equivalents (for 5-bit non-negative integers):

0 = 00000	8 = 01000	16 = 10000	24 = 11000
1 = 00001	9 = 01001	17 = 10001	25 = 11001
2 = 00010	10 = 01010	18 = 10010	26 = 11010
3 = 00011	11 = 01011	19 = 10011	27 = 11011
4 = 00100	12 = 01100	20 = 10100	28 = 11100
5 = 00101	13 = 01101	21 = 10101	29 = 11101
6 = 00110	14 = 01110	22 = 10110	30 = 11110
7 = 00111	15 = 01111	23 = 10111	31 = 11111