

CS 564 Midterm Exam

Fall 2018

Answers

A: LET'S WRITE SOME QUERIES! [30%]

We will use the following database schema:

Student (stid, firstname, lastname, major)

Class (code, title, semester, year, description)

Enrolled (stid, code, numcredits)

Furthermore, Enrolled.stid is a foreign key referring to Student.stid and Enrolled.code is a foreign key referring to Class.code

1. [10%] Express the following query in SQL: *output the student IDs of the students that have taken at least 30 credits across all classes.*

2. [10%] Express the following query in SQL: *for each class offered in Spring 2018, output the class code and the number of CS majors that took that class.*

3. [10%] Express the following SQL query in Relational Algebra:

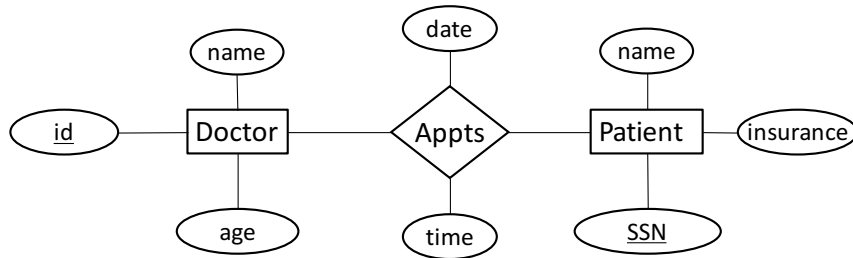
```
SELECT DISTINCT firstname, lastname
FROM Student
WHERE stid NOT IN (
    SELECT s.stid
```

```
FROM Student s, Enrolled e
WHERE s.stid = e.stid AND e.numcredits > 3);
```

B: ER, RELATIONAL MODEL, AND SOME SQL [13%]

For the following questions, circle **exactly one** correct option.

1. [5%] Consider the following ER diagram:



If we translate the relationship **Appts** to a relation in the relational model, how many attributes will the relation have?

2. [4%] Consider the schema from Part A. The following SQL queries will always give the same result:

```
SELECT * FROM Enrolled
LEFT OUTER JOIN Student on Student.stid = Enrolled.stid;
```

```
SELECT * FROM Enrolled
INNER JOIN Student on Student.stid = Enrolled.stid;
```

3. [4%] Consider the following relational table for Student:

studentID	name	age
1234	George	NULL
1144	Anna	19
2214	Maria	NULL

How many tuples will the following SQL query return?

```

SELECT      *
FROM        Student
WHERE       name = 'George' OR age > 20 ;

```

C: NORMALIZATION AND DEPENDENCY THEORY [42%]

Consider the relation $R(A, B, C, D, E)$ with the following set F of functional dependencies:

$A \rightarrow B, C$ $D \rightarrow C$ $E \rightarrow D$ $B, E \rightarrow C$

[10%] Add a tuple to the instance of R below such that **all** fds in F are violated:

A	B	C	D	E
a	b	c	d	e
a	b	c	d'	e'

ANSWER:

For the following questions, circle the right option(s).

#1 There can be more than one correct options for every question!

#2 You can get partial credit by explaining how you came up with your answer.

1. [12%] The following attribute sets are **superkeys** but **not keys** in relation R :

(i) A, E (ii) A, D, E (iii) A, B, C, D (iv) A, B, C, D, E

ANSWER:

2. [8%] The following functional dependencies are **redundant** in F (in other words, if we remove the functional dependency, the fd closure remains the same):

(i) $A \rightarrow B, C$ (ii) $D \rightarrow C$ (iii) $E \rightarrow D$ (iv) $B, E \rightarrow C$

ANSWER:

3. [12%] The following hold for the decomposition of R into ABC, AE, DE . It is:

(i) lossless-join (ii) dependency preserving

ANSWER:

D: BUFFER MANAGEMENT [15%]

In this question, we consider a buffer pool with 5 frames, and two files: one with three pages A_1, A_2, A_3 , and the other with four pages B_1, B_2, B_3, B_4 . We want to read these two files in a nested loop (as we will see later in class, this is one possible implementation of a join between the two files). The sequence of requests is as follows:

Request A_1 , Request B_1 , Release B_1 , Request B_2 , Release B_2, \dots , Release A_1 ,
Request A_2 , Request B_1 , Release B_1 , Request B_2 , Release B_2, \dots , Release A_2 ,
Request A_3 , Request B_1 , Release B_1 , Request B_2 , Release B_2, \dots , Release A_3 .

Initially, all buffer frames are free, and none of the pages to be accessed are in RAM. For the following questions, circle the right option.

#1 You can get partial credit by explaining how you came up with your answer.

1. [10%] If the buffer manager uses the **LRU** replacement policy, how many total I/Os (both reads and writes) occur for the above sequence?

(i) 5 (ii) 7 (iii) 10 (iv) 15

ANSWER:

2. [5%] If the buffer manager knows the sequence a priori, how many total I/Os does the optimal replacement policy achieve?

(i) 0 (ii) 5 (iii) 7 (iv) 10

ANSWER:

BONUS: write your own SQL joke!
