

CSC 343H1S 2012 Midterm Test

Duration — 50 minutes

Aids allowed: none

Student Number:

Last Name:  First Name:

Lecture Section: L0101 (daytime)

Instructor: Horton

---

*Do **not** turn this page until you have received the signal to start.*

(Please fill out the identification section above, **write your name on the back of the test**, and read the instructions below.)

*Good Luck!*

---

This midterm consists of 4 questions on 8 pages (including this one). *When you receive the signal to start, please make sure that your copy is complete.* If you use any space for rough work, indicate clearly what you want marked.

# 1:  / 4

# 2:  / 4

# 3:  / 15

# 4:  / 7

TOTAL:  / 30

---

**Question 1.** [4 MARKS]

Consider the following schema for a database about transit routes, stops along the routes, and the scheduled time to reach the stops on a route (for various trips along the route).

**Relations**

- Route(name, num, mode)
- Stop(route, stopNum, location)
- StopTime(route, stopNum, tripNum, time)

**Integrity constraints**

- $\Pi_{mode}Route \subseteq \{\text{"subway"}, \text{"streetcar"}, \text{"bus"}, \text{"LRT"}\}$
- $Stop[route] \subseteq Route[num]$
- $StopTime[route, stopNum] \subseteq Stop[route, stopNum]$

Which of the following statements are enforced by the schema? Circle one answer for each. If the statement is enforced, say what part of the schema enforces it. If it is not enforced, write an integrity constraint that would enforce it, using one of the two forms defined in the textbook.

(No penalty for wrong answers, but no marks without a correct explanation or integrity constraint.)

1. Two routes cannot have the same name.

Enforced                      This part of the schema enforces it:

Not enforced                This new integrity constraint would enforce it:

2. Every value for route in relation StopTime occurs as a value for num in relation Route.

Enforced                      This part of the schema enforces it:

Not enforced                This new integrity constraint would enforce it:

**Question 2.** [4 MARKS]

Suppose we have two relations: Patient(PID, height) and Caresfor(PID, doctor). Consider the following instance of that schema:

Patient	PID	height	Caresfor	PID	doctor
	12	11		33	44
	25	12		20	44
	96	11		25	30
	20	12		12	30
	33	33		20	18
	44	8		96	30
				12	20

**Part (a)** [2 MARKS]

Give the result (schema and data) returned by the following query. Use the same tabular format as above.

$$Temp(PID, height) := \Pi_{P1.PID, P1.height}(\sigma_{P1.height > P2.height}(\rho_{P1}(Patient) \times \rho_{P2}(Patient)))$$

$$Temp2(PID, height) := \Pi_{T1.PID, T1.height}(\sigma_{T1.height > T2.height}(\rho_{T1}(Temp) \times \rho_{T2}(Temp)))$$

$$\Pi_{PID, doctor}(Temp - Temp2) \bowtie Caresfor$$
**Part (b)** [2 MARKS]

Describe what this query computes. Do not describe the steps it takes, only what is in the result, and make your answer general to any instance of the schema.

**Question 3.** [15 MARKS]

We used the following schema many times in lecture:

**Relations**

- Students(SID, surName, campus)
- Courses(CID, cName, WR)
- Offerings(OID, CID, term, instructor)
- Took(SID, OID, grade)

**Integrity constraints**

- Offerings[CID]  $\subseteq$  Courses[CID]
- Took[SID]  $\subseteq$  Students[SID]
- Took[OID]  $\subseteq$  Offerings[OID]

**Part (a)** [6 MARKS]

Write a query in relational algebra to report the surname and SID of all students who have taken at most one course in which no one in that offering of the course earned a grade of 100. Use only the basic operators  $\Pi, \sigma, \bowtie, \times, \cap, \cup, -, \rho$ .

**Part (b)** [1 MARK]

Can the following query be expressed using those same basic operators ( $\Pi, \sigma, \bowtie, \times, \cap, \cup, -, \rho$ ):  
Report the closest grade to 50 among the passing grades in any offering of csc363. Circle one answer.  
(One mark for a correct answer and -0.5 for an incorrect answer.)

YES

No

**Part (c)** [8 MARKS]

Consider all the students who've ever taken csc333. Suppose we want to find the instructors who have given every one of them a passing grade in at least one course. (They need not have been in the same offering or course.) Which of the following syntactically legal queries will report that?  
(2 marks for each correct answer, -1 for each incorrect answer.)

1.  $Takers(SID) := \Pi_{SID} \sigma_{CID="csc333"}(Took \bowtie Offering)$   
 $Did(SID, instructor) := \Pi_{SID, instructor} \sigma_{grade \geq 50}(Took \bowtie \sigma_{CID="csc333"} Offering)$   
 $(\Pi_{instructor} Offering) - (\Pi_{instructor} [(Takers \bowtie \Pi_{instructor} Offering) - Did])$

Correct          Incorrect

2.  $Takers(SID) := \Pi_{SID} \sigma_{CID="csc333"}(Took \bowtie Offering)$   
 $Did(SID, instructor) := \Pi_{SID, instructor} \sigma_{grade \geq 50}(Took \bowtie Offering)$   
 $(\Pi_{instructor} Offering) - (\Pi_{instructor} [(Takers \bowtie \Pi_{instructor} Offering) - Did])$

Correct          Incorrect

3.  $Takers(SID) := \Pi_{SID} \sigma_{CID="csc333"}(Took \bowtie Offering)$   
 $Did(SID, instructor) := \Pi_{SID, instructor} ([\sigma_{grade \geq 50} Took] \bowtie [\Pi_{OID, instructor} Offering])$   
 $(\Pi_{instructor} Offering) - (\Pi_{instructor} [(Takers \bowtie \Pi_{instructor} Offering) - Did])$

Correct          Incorrect

4.  $Takers(SID) := \Pi_{SID} \sigma_{CID="csc333"}(Took \bowtie Offering)$   
 $Did(SID, instructor) := \Pi_{SID, instructor} ([\sigma_{grade \geq 50} Took] \bowtie [\Pi_{OID, instructor} Offering])$   
 $(\Pi_{instructor} Offering) - (\Pi_{instructor} [Takers \bowtie \Pi_{instructor} Offering] - \Pi_{instructor} Did)$

Correct          Incorrect

**Question 4.** [7 MARKS]

Consider the following schema about athletes and their results in the long jump event:

```
-- Long jump results. "who" is the athlete whose
-- result it is, "distance" is how far they jumped, and
-- "t" is when they did it.
create table longjump (
    who int,
    distance float not null,
    t timestamp,
    primary key (who, t),
    foreign key (who) references athlete(aID)
);
create table athlete (
    aID int primary key,
    name text not null
);
```

**Part (a)** [3 MARKS]

Complete the **where** condition in the following SQL query so that it reports the aID and name of the athlete who jumped the greatest distance. If there is a tie, report them all. If there are duplicates, report them all.

```
select aid, name
from longjump natural join athlete
where
```

**Part (b)** [1 MARK]

Could any query that begins as stated above actually produce duplicate tuples?  
(One mark for a correct answer and -0.5 for an incorrect answer.)

YES                      NO

**Part (c)** [3 MARKS]

Suppose we have a relation R(a, b, c). Consider this query:

```
select _____ from R group by b;
```

Which of the following could go in the select list for this query? Circle Okay or Error for each.  
(One mark for each correct answer and -0.5 for each incorrect answer.)

max(a)	<input type="radio"/> OKAY	<input type="radio"/> ERROR
c	<input type="radio"/> OKAY	<input type="radio"/> ERROR
min(c)	<input type="radio"/> OKAY	<input type="radio"/> ERROR

*[Use the space below for rough work. This page will not be marked unless you clearly indicate the part of your work that you want us to mark.]*

**Last Name:** \_\_\_\_\_ **First Name:** \_\_\_\_\_