Do not turn this page until you have received the signal to start. (In the meantime, please fill out the identification section above, and read the instructions below.)

This test consists of 4 questions on 8 pages (including this one). When you receive the signal to start, please make sure that your copy of the quiz is complete.

If you use any space for rough work or need to scratch out an answer, circle the part that you want us to mark.

Good Luck!
Question 1. [11 marks]

Recall this schema, which we have used many times in class. Here we are adding one more relation called Program. It records the subject POSDs that students are enrolled in. (“POSt” is short for “program of study”, by the way.)

Relations

Student(sID, surName, firstName, campus, email, cgpa)
Course(dept, cNum, name, breadth)
Offering(oID, dept, cNum, term, instructor)
Took(sID, oID, grade)
Program(sID, POSt)

Integrity constraints

Offering[dept, cNum] ⊆ Course[dept, cNum]
Took[sID] ⊆ Student[sID]
Took[oID] ⊆ Offering[oID]
Program[sID] ⊆ Student[sID]

Part (a) [7 marks]

Write a query to find the sIDs of students on campus ‘StG’ who have exactly one subject POSt. Use only the basic operators Π, σ, △◁, ×, ∩, ∪, −, ρ, and assignment.
Part (b)  [4 marks]

Consider the following query:

\[
Croom(instructor, cNum) := (\Pi_{instructor} \text{Offering}) \times (\Pi_{cNum} \sigma_{dept='CSC'} \land cNum \geq 200 \land cNum < 300 \text{Course})
\]

\[
Flep(instructor, cNum) := Croom - (\Pi_{instructor, cNum} \text{Offering})
\]

\[
Answer(instructor) := [(\Pi_{instructor} \text{Offering}) - (\Pi_{instructor} \text{Flep})] - (\Pi_{instructor} \sigma_{dept='MAT'} \text{Offering})
\]

1. Below are instances of the relations that are relevant to this query. Add exactly 5 rows to \textit{Offering} so that professors Able and Bland will not appear in the result of the query, but professors Cranky and Devlish will. Use only these instructor names, and be sure the instance you create is valid.

| Course: | Offering: |
|-----|-----|-----|-----|-----|-----|-----|
| dept | cNum | name | breadth | oID | dept | cNum | term | instructor |
| CSC  | 108  | Intro Prog | false  | 4   | CSC  | 108  | t1   | Cranky |
| CSC  | 207  | Intro Design | true  | 1   | CSC  | 108  | t2   | Able   |
| CSC  | 209  | Sys Prog  | false  | 7   | CSC  | 209  | t1   | Devlish |
| MAT  | 137  | Calculus  | false  | 2   | CSC  | 209  | t1   | Bland  |
| CSC  | 369  | Op Sys    | false  | 9   | CSC  | 207  | t3   | Bland  |
| ENG  | 244  | Shakespeare | true  |      |       |       |      |        |

2. What does this query compute? Do not describe the steps it takes, only what is in the result, and make your answer general to any instance of the schema.

All instructors who ...
Question 2. [6 marks]

Part (a) [2 marks]

At UofT, a student may have no POSt, one POSt, or several POSts. In the previous question, we introduced a new relation called Program to record information about students’ POSts. Instead of making a separate Program relation, we could add a column for POSt in the Student relation. Would that be a good design? Circle one answer:

Yes  No

Explain:

Part (b) [4 marks]

Consider this schema:

\[
\begin{align*}
R(\text{one}, \text{two}, \text{three}) & \quad S[\text{five}, \text{six}] \subseteq R[\text{one}, \text{two}] \\
S(\text{four}, \text{five}, \text{six}) & \quad S[\text{four}] \subseteq T[\text{eight}] \\
T(\text{seven}, \text{eight}) &
\end{align*}
\]

Suppose relation \( S \) has 100 tuples. How many tuples could \( R \) have? Circle all answers that do not violate the schema.

\[
\begin{array}{cccccc}
0 & 1 & 82 & 100 & 101 \\
\end{array}
\]

Suppose relation \( S \) has 100 tuples. How many tuples could \( T \) have? Circle all answers that do not violate the schema.

\[
\begin{array}{cccccc}
0 & 1 & 82 & 100 & 101 \\
\end{array}
\]
Question 3.  [5 marks]

The question refers to the schema from Question 1. Write a query in SQL to find the total number of courses in the ‘CSC’ department each student has taken. Report the student id and the total number of distinct ‘CSC’ courses taken.
Question 4. [8 marks]

Part (a) [3 marks]

Consider the same schema from the Question 1. Suppose we wrote the query

```
SELECT ________________________
FROM Offering, Took
WHERE Offering.oID = Took.oID
GROUP BY Offering.oID;
```

Which of the following could go in the `SELECT` clause? Circle all that apply.

```
sID  count(sID)  Offering.oID  grade  avg(grade)  count(instructor)  oID
```

Part (b) [3 marks]

We discussed in lecture how SQL subquery operators could possibly be implemented using other SQL operations. Suppose we have two tables R(a,b) and S(b,c).

Consider the following two queries:

```
-- Query 1
SELECT a AS answer
FROM R
WHERE EXISTS
(SELECT * FROM S
  WHERE c > a AND R.b = S.b);

-- Query 2
SELECT R.a AS answer
FROM R, S
WHERE c > a AND R.b = S.b;
```

On the next page, give a database instance where these two queries produce different results, and the results of the two queries.
Instance of the database where the two queries produce different results:

Result of Query 1 on this instance: 

Result of Query 2 on this instance: 

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

Here is an attempt to fix Query 2, to make it identical to Query 1. Explain, in English, why this new query is not the same as Query 1. Don't just give a database instance – write a sentence or two describing the problem precisely.

-- Query 3

```
SELECT DISTINCT R.a AS answer
FROM R, S
WHERE c > a AND R.b = S.b;
```
This page is left (mainly) blank for things that don’t fit elsewhere.

# 1: _____/11
# 2: _____/ 6
# 3: _____/ 5
# 4: _____/ 8

TOTAL: _____/30