CSE 344 – 26wi – Midterm Exam Preview

Last Name:		
First Name:		
UWNetID	@uw.edu	
Person to your Left | Person to your Right	@uw.edu	@uw.edu
All submitted work is my own. I will not share the contents of this exam with others who have not taken it yet. *I acknowledge that violation of these terms could result in a failing grade* **(please sign)**		

Instructions:

- This exam contains **8 pages**, including this cover page. Show scratch work for partial credit, but write your final answers in the boxes and blanks provided.
- Write your UWNetID at the **top of every page**. This helps us reassemble your exam when its pages are inevitably separated and lost.
- The exam is **closed book**. Please do not use laptops, tablets, wearable devices, mobile phones, notes sheets, or books.
- Please silence and put away all noise-making devices, including mobile phones and wearable devices.
- You have 50 minutes to complete this exam.

Advice:

- Read *all* the questions before answering any questions. Start where you feel the most confident.
- Skip questions that are taking a long time. You can always come back.
- Breathe. You've got this. You belong here.

Question	**1**	**2a-2b**	**3a-3b**	**3c**	**3d-3e**
Possible Points	XXX	XXX	XXX	XXX	XXX

UWNetID: _____@uw.edu

Question 1 (XXX points):

Assume we have the following relation:

```
CREATE TABLE R(
  aa  VARCHAR(256),     -- nullable!
  bb  VARCHAR(256),     -- nullable!
  i   INTEGER NOT NULL  -- values will always be non-negative
);
```

and an instance of `R` which contains **17 rows** total. Across the entire table:

- There are a total of **2 distinct values for `aa`**
  - These values are '`foo`' and '`bar`'
- There are a total of **5 distinct values for `bb`**
  - Four of them are non-empty and the last value is the empty string
- There are **17 distinct values for `i`**
  - In other words, every row in the table is unique

When we consider the rows contained in this specific instance:

- There are **11 rows** with an `aa` value of '**`foo`**'
  - 4 of these rows each have a different non-empty value for `bb`
  - The remaining 7 rows all share the empty string for `bb`
- There are **6 rows** with an `aa` value of '**`bar`**'
  - All 6 of these rows all share the same non-empty string for `bb`

Finally, consider the following query:

```
  SELECT r1.aa, r1.bb
    FROM R AS r1,
         R AS r2
   WHERE r1.aa = r2.aa
GROUP BY r1.aa, r1.bb, r1.i
  HAVING r1.i = MAX(r2.i)
ORDER BY r1.bb;
```

How many tuples or groups are output by each phase of the previous query's execution?

JOIN	tuples
WHERE	tuples
GROUP-BY	groups
HAVING	groups
ORDER BY	groups
SELECT	groups

For your convenience, we have summarized the previous page's information here:

```
R(aa, bb, i);

  SELECT r1.aa, r1.bb
    FROM R AS r1,
         R AS r2
   WHERE r1.aa = r2.aa
GROUP BY r1.aa, r1.bb, r1.i
  HAVING r1.i = MAX(r2.i)
ORDER BY r1.bb;
```

- **2** distinct values for **aa**
- **5** distinct values for **bb**
- **17** distinct values for **i**

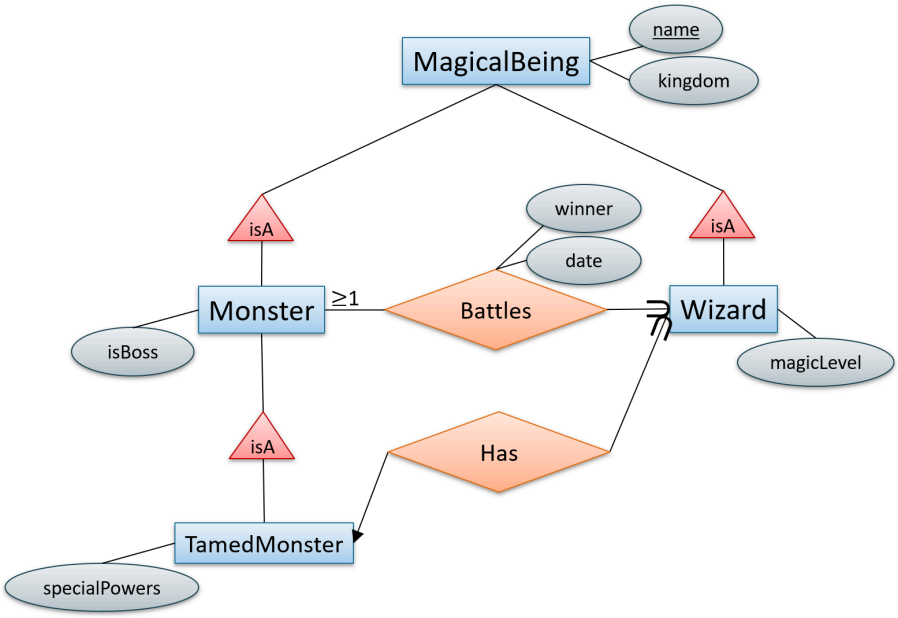
17 rows total

- **11 rows** with `aa='foo'`
  - `bb` values vary, see above
- **6 rows** with `category='bar'`
  - `bb` values are identical

UWNetID: _____@uw.edu

Question 2:

Consider the following ER diagram:



Question 2.a (XXX points):

If this diagram were converted to SQL, which of the following **can be represented as a set of columns** in a table corresponding to a differently-named entity set? Select all that apply.

- ☐ MagicalBeing
- ☐ Monster
- ☐ TamedMonster
- ☐ Battles (between Monster and Wizard)
- ☐ Has (between TamedMonster and Wizard)
- ☐ None of the above statements are true

Question 2.b (XXX points):

Consider a query that performs a join between two tables: `Monsters` and `Wizards`. Let its join clause be:

```
Monsters INNER JOIN Wizards ON Monsters.name = Wizards.opponentName
```

where **`Monsters.name` is the primary key** in `Monsters` and there are **no restrictions** on the values which can be stored in `Wizards.opponentName`. (i.e., `Wizards` may contain "orphaned keys"). Select all the statements which are **always true**.

- ☐ Every row from `Monsters` appears at least once in the query result
- ☐ Every row from `Wizards` appears at least once in the query result
- ☐ The query result may contain rows with `Monsters.name=NULL`
- ☐ The query result may contain rows with `Wizards.opponentName=NULL`
- ☐ None of the above statements are true

UWNetID: _____@uw.edu

Question 3:

Consider the relation:

```
Treats(Brand, Name, Price, FillingType, CacaoPercentage, Texture)
```

You may assume the following functional dependencies exist on its attributes.

1. `Brand → Texture`
2. `CacaoPercentage, Texture → FillingType`
3. `Brand, Texture → Name`

Question 3.a (**XXX** points):

{FillingType}$^{+}$ = _____

Question 3.b (**XXX** points):

{Brand}$^{+}$ = _____

Question 3.c (XXX points):

Decompose `Treats` into BCNF. At each step of your decomposition, state which "bad FD" you are decomposing on. When you are done, circle the final relations. For each of your final relations, annotate which of the original FDs apply to it and indicate its keys.

For your convenience, we have summarized the previous page's information here:

```
Treats(Brand, Name, Price, FillingType, CacaoPercentage, Texture)
    1. Brand → Texture
    2. CacaoPercentage, Texture → FillingType
    3. Brand, Texture → Name
```

Question 3.d (XXX points):

As a reminder, `Treats` is defined as:

```
Treats(Brand, Name, Price, FillingType, CacaoPercentage, Texture)
```

Assume you have a **specific instance** of `Treats`, which consists of **128 rows** that you gathered from a visit to your local grocery store. Here are the exact values of **one of the rows** in that instance.

```
('Hersheys', 'King-sized Milk Chocolate', 2.99, 'None', 11, 'Chalky')
```

Your friend looks over your shoulder and immediately comments "Huh, `CacaoPercentage → Price` holds on your dataset". Since she's very knowledgeable about data management, which of the following statements **must be true**? Select all that apply.

- ☐ If there is a row with `CacaoPercentage=11`, then its `Price` **must be** `2.99`
- ☐ If there is a row with `CacaoPercentage=11`, then its `Price` **may be** `2.99`
- ☐ If there is a row with `Price=2.99`, then its `CacaoPercentage` **must be** `11`
- ☐ Adding a 129th row with values
  `('Lindt', 'Milk Chocolate', 2.99, 'None', 31, 'Smooth')`
  would cause the `CacaoPercentage → Price` **FD to no longer hold**
- ☐ None of the above statements are true

Question 3.e (XXX points):

You go to a different grocery store and gather an additional 64 rows for your dataset. Unfortunately, your friend is gone and you can't ask whether `CacaoPercentage → Price` still holds on your new instance. Select all of the statements that **must be true**.

- ☐ `CacaoPercentage → Price` **no longer holds** over the 192=128+64 rows of our new instance
- ☐ If `CacaoPercentage → Price` **still holds**, then the 64 new rows **must not include**
  `('Lindt', 'Hazelnut White Chocolate', 5.00, 'Hazelnut', 11, 'Crunchy')`
- ☐ If `Price → Brand` **no longer holds**, then the 64 new rows **must include** a row with `CacaoPercentage=11` and `Price=2.99`
- ☐ None of the above statements are true