### Final-Exam

Name: Yue Lin

CWID: 10479231

1.

I think that all these four assumptions are reasonable. Because the given data just told us the instances of A and B, not the definite relations and entities, for example teachers and students.

So explanations for relation1( 1: N)

Assume that A is teacher, B is the course. One teacher can teach n courses.

explanations for relation2( N: 1)

Assume that A is students, B is the classroom, different students can exist in a classroom.

explanations for relation3( N: M)

Assume that A is teacher, B is the course. Because there are no too many rows of the given data. We can also think that one teacher can teach different courses and one course can be taught by different teachers.

explanations for relation4( 1: 1)

We can see the given data, each B instances is R’ed by A. So assume that A is school and B is headmaster, one school have one headmaster.

2.

We notice that there are two different underline in these three entities, one is solid line and another is dotted line. Solid line means identifying relationship and dotted line means non-identifying relationship.

Therefore, for set A, its key is a3, because {a3} has solid line.

Because in an identifying relationship, the primary key is part of the foreign key. So for set B, its keys are {a3,b4}.

The same reason for set C, it needs to combine with set B, and its keys are {a3,b4,c4}.

3.

The attribute c5 in set A is combined by the attributes {c1,c2,c3}, so I put these three attributes into set A directly without the connection of c5.

First initial translation of A :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| a1 | a3 | a4 | c2 | c3 |

As shown, set A and set B’ relation is 1 to 1, so I can not combine A and B into same set. And B’s primary key is {b1}, I will add {b1} attribute to A schema as FKR.

Therefore, the Minimal Storage Implementation is:

A

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| a1 | a3 | a4 | c1 | c2 | c3 | b1 |

B

|  |  |  |  |
| --- | --- | --- | --- |
| b1 | b2 | b3 | b4 |

4.

As shown of the question, set A has four attributes {a1,a2,a3,a4} and set B has {b1,b2}, and a1 is the primary key of A, b1 is the primary key of B. And their relation is 1 to N. So firstly I will create a table to store to two tables’ key to connect them.

Table A\_B

|  |  |
| --- | --- |
| a1 | b1 |

And one A can have many B rows, so I can combine table B and table A\_B. The relation why I combine these two tables instead of A and A\_B is that when the relation is 1 to N, we often merge table from many sides and the primary key is also from the M table.

Therefore, the new combined table and table A:

B\_Relation:

|  |  |  |
| --- | --- | --- |
| b1 | b2 | a1 |

A

|  |  |  |  |
| --- | --- | --- | --- |
| a1 | a2 | a3 | a4 |

5.

In this picture, set A has four attributes {a1,a2,a3,a4} and set B has {b1,b2}, and a1 is the primary key of A, b1 is the primary key of B. And their relation is N to M. So firstly I will create a table, that holds FKRs to the key of the set A and set B table.

Table A\_B

|  |  |
| --- | --- |
| a1 | b1 |

Because the relation is N to M, I can not combine them into a table, just use another table to connect. So the minimal storage design is the below.

Table A\_B

|  |  |
| --- | --- |
| a1 | b1 |

A

|  |  |  |  |
| --- | --- | --- | --- |
| a1 | a2 | a3 | a4 |

B

|  |  |
| --- | --- |
| b1 | b2 |

6

Consider the following table schema and functional

dependencies

-

Table 2 ( Y, X, W, V, U, T)

-

Z -->Y

-

TY -->X

-

TY -->V

-

TY -->U

-

X -->W

-

W -->V

Find a key for Table 2 and show that it’s actually a key (No credit otherwise)