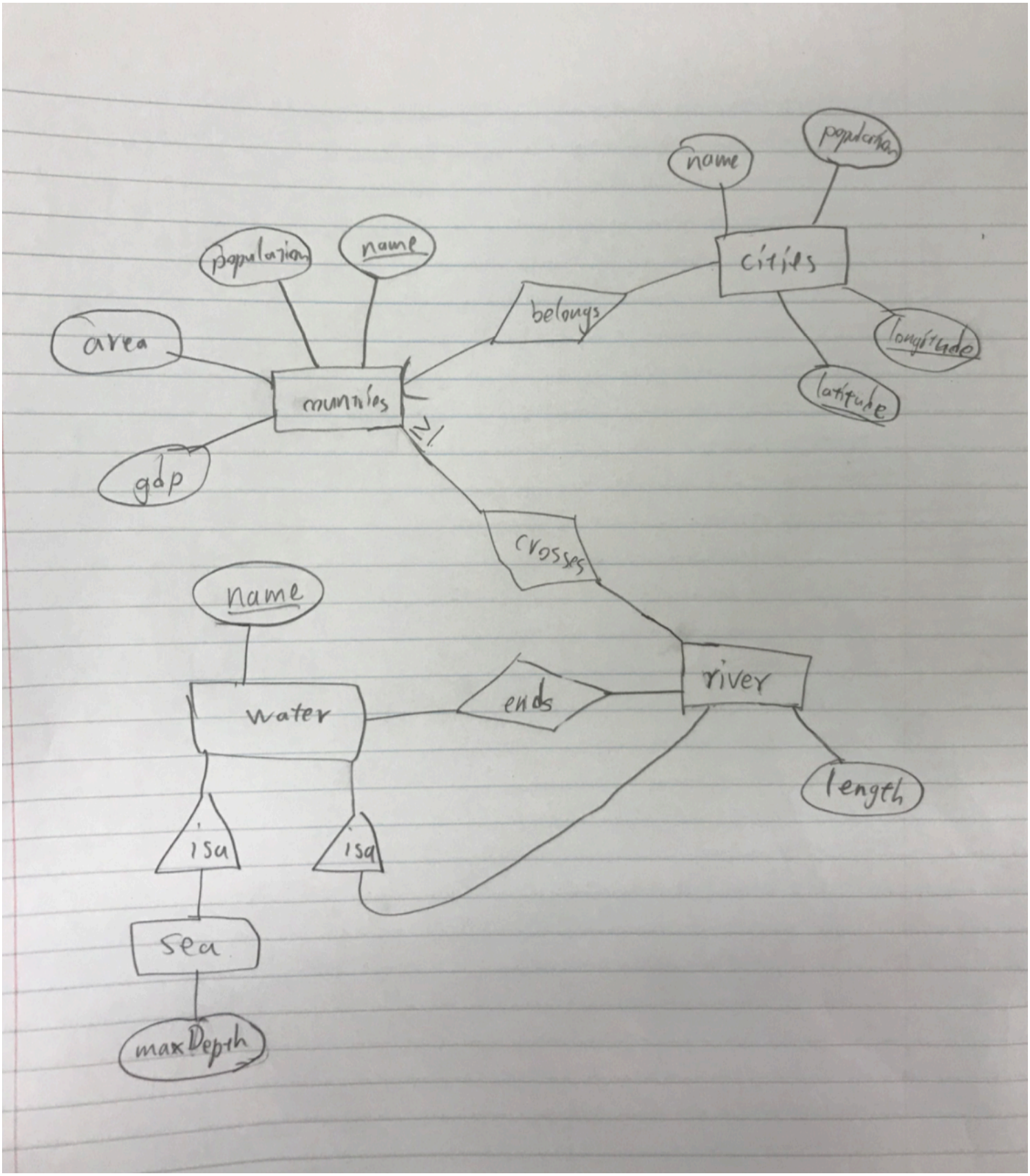
name
population
cities
population
name
belongs
area
longitude
countries
latitude
gdp
≥1
crosses
name
river
water
ends
length
isa
isa
sea
maxDepth

```
CREATE TABLE InsuranceCo(
        name varchar(20) PRIMARY KEY,
        phone int,
);

CREATE TABLE Person(
        ssn int PRIMARY KEY,
        name varchar(20),
);

CREATE TABLE Vehicle(
licensePlate varchar(20) PRIMARY KEY,
year int,
ssn int,
insuranceCoName varchar(20),
maxLiability real,
FOREIGN KEY(InsuranceCoName) REFERENCES InsuranceCo(name),
FOREIGN KEY(ssn) REFERENCES Person(ssn)
);

CREATE TABLE Car(
licensePlate varchar(20) PRIMARY KEY,
        make varchar(20),
FOREIGN KEY(licensePlate) REFERENCES Vehicle(licensePlate)
);

CREATE TABLE Driver(
        ssn int PRIMARY KEY,
        driverID int,
FOREIGN KEY(ssn) REFERENCES Person(ssn),
);

CREATE TABLE NonProfessionalDriver(
ssn INT PRIMARY KEY,
FOREIGN KEY(ssn) REFERENCES Driver(ssn)
);

CREATE TABLE ProfessionalDriver(
ssn INT PRIMARY KEY,
medicalHistory VARCHAR(20),
FOREIGN KEY(ssn) REFERENCES Driver(ssn)
);

CREATE TABLE Truck(
licensePlate varchar(20) PRIMARY KEY,
        capacity int,
        ssn int
```

```
FOREIGN KEY(ssn) REFERENCES ProfessionalDriver(ssn),
FOREIGN KEY(licensePlate) REFERENCES Vehicle(licensePlate)
);


CREATE TABLE Drives(
licensePlate VARCHAR(20),
ssn INT,
FOREIGN KEY(licensePlate) REFERENCES car(licensePlate),
FOREIGN KEY(ssn) REFERENCES NonProfessionalDriver(ssn)
);
```

the relational schema that represents the relationship "insures" is the foreign key connect from each vehicle to InsuranceCo, the vehicles contain the attribute "maxLiability" and also the primary key of InsuranceCo. Because the relationship between vehicles and InsuranceCo is many-one, for each vehicles can have at most one InsuranceCo.
The representation of the relationships "drives" and "operates" in my schema are different, "drives" relationship is between entity Car and NonProfessionalDriver. It's a many to many relationship, which means each car and have more than one drivers, and each driver can nonprofessional drive more than one car. While the "operates" relationship between entity Truck and ProfessionalDriver is many-one relationship, which means each professional driver can drive at most one truck, each truck can be operates by at most one professional driver.

/* Question 3 */
R(A, B, C, D, E) with FDs D -> B, CE -> A
{D}+ = {D, B} thus D is not a super key, BCNF violation
Use D -> B violation to decompose R into R1(D, B) and T1(A, C, D, E) {CE}+ = {C, E, A} thus CE is not a super key, BCNF violation
Use CE -> A violation to decompose T1 into R2(C, E, A) and R3(C, E, D) Final Relationships: R1(D, B), R2(C, E, A) and R3(C, E, D)
S(A, B, C, D, E) with FDs A -> E, BC -> A, DE -> B
{A}+ = {A, E} thus A is not a super key, BCNF violation
Use A -> E violation to decompose S into R1(A, E) and T1(A, B, C, D)
————————————————————
{BC}+ = {B, C, A} thus BC is not a
DE -> B relationship disappear, so
Use BC -> A violation to decompose
This is just one way to decompose.
Final Relationships: R1(A, E), R2(*B, C*, A), R3(B, C, D)
super key, BCNF violation
no longer need it
T1 into R2(B, C, A) AND R3(B, C, D)

/* Question 4 */
- All sets of attributes are closed
Since all sets of attributes are closed, this happens when it only has functional dependencies!
Solution FDs: {}
-{}, {A, B, C, D}
Solution FDs: {A -> B, B -> C, C -> D, D -> A}
-{}, {A, B}, {A, B, C, D}
Solution FDs: {C -> ABD, D -> ABC, A -> B, B -> A}