Chapter 4

11. Given the following information, produce an ERD-based on the Crow’s Foot notation- that can be implemented. Make sure you include all appropriate entities, relationships, connectivity, and cardinalities.

```
   customer
     | has / is of
     |      |
     |      v
     |      UPDATE
     v
   CAR
     | has / is of
     |      
     |      MAINTENANCE
     |      | has / performed on
     v     
   INVOICE
     | has / is of
     |      
     |      MAINT_LINE
     |      | includes / is found in
     v     
   PART
     | is ordered from / fulfills order
     |      
     |      SUPPLIER

  pays / given to
```

12. a. Given that brief and incomplete description of operations for RC_Models Company, write all applicable business rules to establish entities, relationships, optionalities, connectivities, and cardinalities. *(Hint: Use the following three business rules as examples, writing the remaining business rules in the same format.)*

- A customer may generate many invoices.
- Each invoice is generated by only one customer.
- Some customers have not (yet) generated and invoice.
- One company sells many products
- Many products are sold by one company
- Many customers pay by only credit card (does not define if a customer can pay by many credit cards)
- Each credit card payment is generated by one customer
- Each invoice has many products
- One or more products are printed on each invoice
- Zero to many products are tracked in inventory by date last used
- Inventory has zero to many products
- Products in inventory less then 4 weeks are kept
- Products in inventory greater then 4 weeks are scrapped
- Inventory has zero to many orders
- Orders may or may not be placed depending on quantizes
- Products have one or more manufactures
- Manufactures may or may not have supplied parts

b. Draw the fully labeled and implementable Crow’s Foot ERD based on the business rules you wrote in Part (a) of this problem. Include all entities, relationships, optionalities, connectivities, and cardinalities.
15. Given the table structure illustrated in Table P5.15, write the relational schema and draw its dependency diagram. Label all transitive and/or partial dependencies. (Hint: This structure uses a composite primary key).

First Normal Form (1NF)
Dependency Diagram

Partial dependency

Partial dependency

Partial dependency

Transitive dependency

Partial dependency

Partial dependency

Partial dependency

Partial dependency

1NF:
(MEMBER_NUM, INVITE_NUM, MEMBER_NAME, MEMBER_ADDRESS, MEMBER_CITY, MEMBER_ZIPCODE, INVITE_DATE, ACCEPT_DATE, DINNER_DATE, DINNER_ATTEND, DINNER_CODE, DINNER_CODE, ENTREE_CODE, DINNER_DESCRIPTION, DESSERT_CODE, DESSERT_DESCRIPTION, ENTREE_DESCRIPTION)
DINNER_DESCRIPTION, ENTÉE_CODE, ENTÉE_DESCRIPTION, DESSERT_CODE, DESSERT_DESCRIPTION)

Partial dependencies:
1. (MEMBER_NUM -> MEMBER_NAME, MEMBER_ADDRESS, MEMBER_CITY, MEMBER_ZIPCODE)
2. (INVITE_NUM -> INVITE_DATE, ACCEPT_DATE, DINNER_DATE)
3. (DINNER_CODE -> DINNER_DESCRIPTION, ENTÉE_CODE, ENTÉE_DESCRIPTION, DESSERT_CODE, DESSERT_DESCRIPTION)

Transitive dependency:
1. (DINNER_DATE -> INVITE_DATE)

16. Break up the dependency diagram you drew in Problem 15 to produce dependency diagrams that are in 3NF and write the relational schema. (Hint: You might have to create a few new attributes. Also make sure that the new dependency diagrams contain attributes that meet proper design criteria; that is, make sure that there are not multi-valued attributes, that the naming conventions are met, and so on.)

(MEMBER_NUM, INVITE_NUM, MEMBER_NAME, MEMBER_ADDRESS, MEMBER_CITY, MEMBER_ZIPCODE, INVITE_DATE, ACCEPT_DATE, DINNER_DATE, DINNER_ATTEND, DINNER_CODE, DINNER_DESCRIPTION, ENTÉE_CODE, ENTÉE_DESCRIPTION, DESSERT_CODE, DESSERT_DESCRIPTION)

MEMBER (MEMBER_NUM, MEMBER_NAME, MEMBER_ADDRESS, MEMBER_CITY, MEMBER_ZIPCODE)
INVITE (INVITE_NUM, MEMBER_NUM, ACCEPT_DATE, DINNER_DATE, DINNER_ATTEND)
DINNER (DINNER_CODE, DINNER_DESCRIPTION, ENTÉE_CODE, DESSERT_CODE)
ENTÉE (ENTÉE_CODE, ENTÉE_DESCRIPTION)
DESSERT (DESSERT_CODE, DESSERT_DESCRIPTION)
SCHEDULE (SCH_NUM, MEMBER_NUM, INVITE_NUM, DINNER_CODE, MEMBER_STATUS)
DATE (DINNER_DATE, INVITE_DATE)
Using the results of Problem 16, draw the Crow’s Foot ERD.
21. Given the sample records in the CHARTER table shown in Table P5.21, write the relational schema and draw the dependency diagram for the table structure.

First Normal Form (1NF)

Dependency Diagram

1NF:

(CHAR_TRIP, CUST_NUM, AC_NUMBER, CHAR_DATE, CHAR_CITY, CHAR_MILES, CUST_LNAME, CHAR_PAX, CHAR_CARGO, PILOT, COPILOT, FLT_ENGINEER, LOAD_MASTER, AC_NUMBER, MODEL_CODE, MODEL_SEATS, MODEL_CHG_MILE)
Partial dependencies:
1. (CHAR_TRIP-> CHAR_DATE, CHAR_CITY, CHAR_MILES, CHAR_PAX, CHAR_CARGO)
2. (CUST_NUM-> CUST_LNAME)
3. (AC_NUMBER-> MODEL_CODE, MODEL_SEATS, MODEL_CHG_MILE)
4. (CHAR_TRIP -> PILOT, COPILOT, FLT_ENGINEER, LOAD_MASTER)

22. Decompose the dependency diagram you drew to solve Problem 21 to create table structures that are in 3NF and write the relational schema. Make sure that you label all dependencies.

(CHAR_TRIP, CUST_NUM, AC_NUMBER, CHAR_DATE, CHAR_CITY, CHAR_MILES, CUST_LNAME, CHAR_PAX, CHAR_CARGO, PILOT, COPILOT, FLT_ENGINEER, LOAD_MASTER, AC_NUMBER, MODEL_CODE, MODEL_SEATS, MODEL_CHG_MILE)

TRIP (CHAR_TRIP, CHAR_DATE, CHAR_CITY, CHAR_MILES, CHAR_PAX, CHAR_CARGO, 
CREW ( CHAR_TRIP, AC_NUMBER, CREW_PILOT, CREW_COPILOT, CREW_FLT_ENGINEER, CREW_LOAD_MASTER)
AIRPLANE (AC_NUMBER, MODEL_CODE, MODEL_SEATS, MODEL_CHG_MILE)

23. Draw the Crow's Foot ERD to reflect the properly decomposed dependency diagrams you created in Problem 22. Make sure that the ERD yields a database that can track all of the data shown in Problem 21. Show all entities, relationships, connectivities, optionalities, and cardinalities.