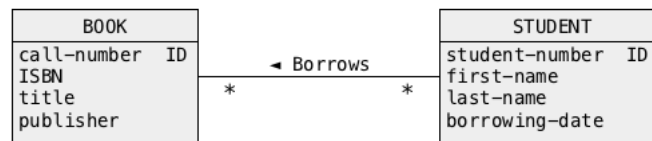


### Task 1 (6 marks)

#### Normalization of relational tables

Consider the following conceptual schema of a sample database domain where the students borrow the books from a library. A book is described by a call-number that uniquely identifies each copy of a book kept in a library. A library purchases several identical textbooks for the subjects enrolled by the larger number of students. A book is identified by ISBN and it is described by a title and publisher.

The students are described by a student number, first name and last name. A date is recorded when a student borrows a book.



A database designer made few mistakes at both conceptual modelling stage and logical design stage (i.e. transformation of a conceptual schema into the relational schemas).

At present, the relational schemas created by a database designer are the following.

BOOK(call-number, ISBN, title, publisher)  
primary key = (call-number)

STUDENT(student-number, first-name, last-name)  
primary key = (student-number)

BORROW(call-number, ISBN, student-number, borrow-date)  
primary key = (call-number, student-number)

Your task is to use the analysis of functional dependencies and normalization of relational schemas to find the highest normal form valid for each one of the relational schemas listed above. If a relational schema is not in BCNF then you must decompose it into the relational schemas in BCNF.

The process of normalization of relational schemas must be performed in the following way. First, find functional dependencies valid in a relational schema. Next, find the minimal keys. Next find, the highest normal form valid for a relational schema, and finally if a normal form found is not BCNF decompose a schema into BCNF.

Repeat such process for every relational schema listed above.

Please remember, that both conceptual schema and relational schemas are not completely correct, so be very careful when finding functional dependencies. Using a conceptual schema and relational schemas only is like "walking over a mine field".

Finally, please do not send to me emails saying that the design is incorrect, I know that it is incorrect !

```
BOOK(call-number, ISBN, title, publisher)
```

Functional dependencies:

```
call-number → ISBN, title, publisher
```

```
ISBN → title, publisher
```

```
title → ISBN (I could not find any counterexample on Amazon)
```

Minimal key:

```
(call-number)
```

The highest normal form:

2NF, 3NF violated by title being transitively dependent on call-number:

```
call-number → ISBN and ISBN → title
```

Decomposition:

```
BOOKCOPY(call-number, ISBN)
```

```
BOOKINCATALOG(ISBN, title, publisher)
```

---

```
STUDENT(student-number, first-name, last-name)
```

Functional dependencies:

```
student-number → first-name, last-name
```

Minimal key:

```
(student-number)
```

The highest normal form:

BCNF

---

```
BORROW(call-number, ISBN, student-number, borrow-date)
```

Functional dependencies:

```
call-number → ISBN
```

Minimal key:

```
(call-number, student-number, borrow-date)
```

The highest normal form:

1NF, 2NF is violated by `call-number`  $\rightarrow$  `ISBN` where nonprime attribute `ISBN` depends on a subset of minimal key,

Decomposition

`BOOKCOPY'` (`call-number`, `ISBN`)

`BORROW'` (`call-number`, `student-number`, `borrow-date`)

`BOOKCOPY'` is identical as `BOOKCOPY` and it can be eliminated.

---

### **Deliverables**

A file `solution1.pdf` with a report from normalization of the relational schemas given above. A report must include a list of functional dependencies found, complete derivations of minimal keys, complete identification of the highest normal form valid and decomposition into BCNF whenever it is necessary.

---