Dr. Stefan Brass
Institut für Informatik
Martin-Luther-Universität Halle-Wittenberg

04. Juni 2003

Vorlesung "Datenbanken II"
— Zwischenklausur —

Name: ______________________________

Matrikelnummer: ______________________________

Aufgabe	Punkte	Max. Punktzahl
1 (Konzeptioneller Entwurf im ER-Modell)		10
2 (Übersetzung ins Relationenmodell)		10
3 (Relationale Normalformen)		6
Summe		26

Anleitung

- Sie haben 60 Minuten Zeit für diese Klausur. Wir werden ca. um 10^{30} beginnen, so daß der Abgabetermin ca. um 11^{30} ist (genauer Termin wird an die Tafel geschrieben). Sie dürfen erst ca. 10 Minuten vor Ende der Klausur gehen, da danach niemand mehr auf Toilette gehen kann (Es darf immer nur ein Teilnehmer gleichzeitig den Raum verlassen.)
- Bitte schreiben Sie die Antworten in den vorgegebenen Platz. Falls der Platz nicht ausreicht, können Sie weitere Blätter oder die Rückseite verwenden, aber bitte markieren Sie klar, daß es noch eine Fortsetzung gibt. Schreiben Sie Ihren Namen und die Nummer der Aufgabe auf alle zusätzlichen Blätter.
- Bitte fragen Sie, wenn Sie die Aufgabe nicht verstehen, oder die Aufgabe Ihnen mehrdeutig erscheint.
- Die Klausur hat 14 Seiten. Bitte prüfen Sie die Vollständigkeit.
- Bitte nehmen Sie keine Klausur mit und auch keine Notizen über die Aufgaben. Sie bekommen die Klausur zurück, sobald sie korrigiert ist.
- Sie können alle schriftlichen Aufzeichnungen verwenden, die vor Beginn der Klausur entstanden sind. Dagegen sind Computer (Laptops etc.) leider nicht zulässig. Es ist auch nicht viel Zeit für langes Suchen in den Aufzeichnungen.
- Bitte vermeiden Sie jedes Verhalten, das als Täuschungsversuch mißverstanden werden könnte. Achten Sie auf einen ausreichenden Abstand zu Ihren Nachbarn und setzen Sie sich nicht direkt neben die Teilnehmer, mit denen Sie vorher immer die Hausaufgaben zusammen abgegeben haben. Schauen Sie nicht in Richtung der Klausuren anderer Teilnehmer und sprechen Sie nicht mit anderen Teilnehmern, bis alle Klausuren abgegeben sind. Tauschen Sie keinesfalls irgendwelche Papiere oder Gegenstände aus. Notfalls rufen Sie bitte den Dozenten zur Kontrolle.
- Bitte stellen Sie sicher, daß ich Ihre Handschrift lesen kann. Verwenden Sie keine rote Tinte.
- Sie dürfen mit Bleistift schreiben, aber es könnte bei späterer Kritik an meiner Korrektur eventuell schwierig werden, zu beweisen, daß Sie etwas geschrieben haben. Dennoch sollten Sie natürlich nachfragen, wenn Sie meine Korrektur nicht verstehen.
- Meine Zeitschätzung ist: Pro Aufgabe 15 Minuten, die Zeitreserve beträgt dann noch 15 Minuten. Falls Sie Schwierigkeiten mit einer Aufgabe haben, bearbeiten Sie zunächst die anderen. Es empfiehlt sich aber, alle Aufgaben wenigstens zu versuchen (wenn Sie nichts hinschreiben, bekommen Sie für die Aufgabe sicher 0 Punkte).
- Viel Erfolg!

Aufgabe 1 (Konzeptioneller Entwurf im ER-Modell) 10 Punkte

Entwicklen Sie ein Datenbankschema im ER-Modell für eine Hausbaufirma mit den unten stehenden Anforderungen. Benutzen Sie dazu die Oracle Designer Notation für das ER-Modell.

- Die Firma hat verschiedene Haustypen. Zu jedem Haustyp sind der Name, Beschreibung (kann null sein), und der Basispreis abzuspeichern. Haustypen werden eindeutig über den Namen identifiziert.
- Außerdem werden Optionen angeboten, wie etwa ein Keller, eine Solaranlage zur Unterstützung der Heißwasserbereitung, Rollläden, Heizkörper mit Handtuchtrocknerfunktion im Bad, eine Zisterne für Regenwassernutzung, und vieles mehr. Optionen werden über eine Nummer identifiziert, haben eine Bezeichnung und eine Beschreibung (kann null sein).
- Für die Optionen ist auch ein Preis abzuspeichern, aber hierbei sind zwei Arten von Optionen zu unterscheiden:
  - Bei manchen Optionen hängt der Preis auch von dem Haustyp ab (z.B. ist der Keller für Häuser mit größerer Grundfläche teurer, und der Preis für die Option "Rollläden an allen Fenstern" hängt von der Fensterzahl ab). Manche dieser Optionen sind auch nicht für alle Haustypen verfügbar. Jede Option ist aber für mindestens einen Haustyp verfügbar.
  - Bei der zweiten Art von Optionen hängt der Preis nicht vom Haustyp ab. Dies gilt etwa für die Zisterne, die Solaranlage, und den Heizkörper mit Handtuchtrocker-Funktion. Für solche Optionen soll der Preis dann auch nur einmal abgespeichert werden, statt einmal für jeden Haustyp.
- Schließlich sind noch Informationen über Hausbau-Projekte abzuspeichern. Hausbau-Projekte werden über ihre Vertragsnummer identifziert. Zusätzliche Daten sind der Name des Kunden, die Adresse des Grundstücks (kann noch unbekannt sein), und der vereinbarte Preis (die Kunden bekommen Rabatte gegenüber der Summe der Einzelpreise, die von ihrem Verhandlungsgeschick und der Auftragslage abhängen.)
- Zu jedem Hausbau-Projekt gehört genau ein Haustyp und null bis beliebige viele Optionen. Es kann Haustypen und Optionen in der Datenbank geben, die bisher noch von keinem Kunden gewählt wurden.
- Soweit nicht ausdrücklich erwähnt, sind die Attribute nicht null.

Falls Ihnen zusätzliche Integritätsbedingungen (Constraints) wichtig erscheinen, die nicht im ER-Diagramm dargestellt sind, geben Sie diese Bedingungen bitte an (informell). Die Datentypen für die Attribute brauchen Sie nicht anzugeben, und damit auch nicht Bedingungen wie "Preise sind immer nicht-negativ."

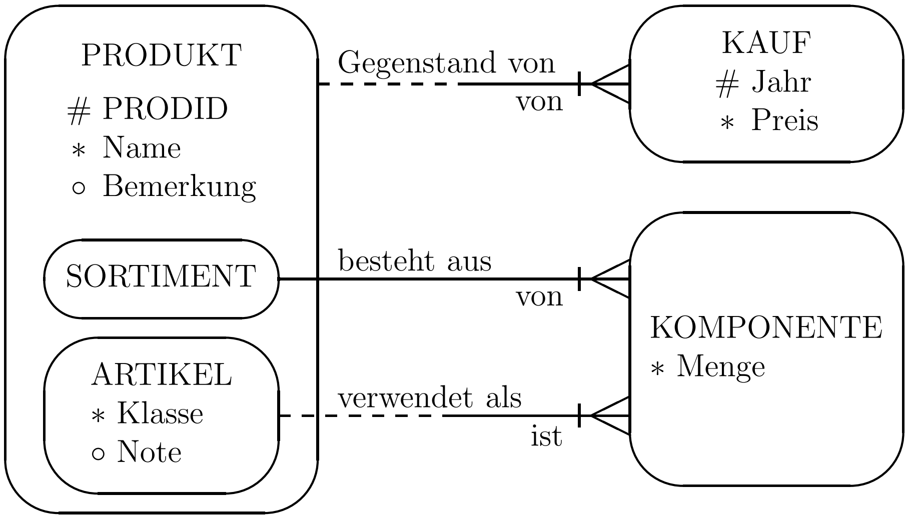
Auf der nächsten Seite ist Platz für Ihre Lösung.

Lösung zu Aufgabe 1

- Haustyp: Name, Beschreibung (kann null sein), Basispreis.
- Option: Nummer, Bezeichnung, Beschreibung (kann null sein).
- Preis bei Optionen: Mal vom Haustyp abhängig, mal nicht.
- Hausbau-Projekt: Vertrags-Nummer, Kunden-Name, Grundstück-Adresse (kann null sein), Preis, genau ein Haustyp, beliebig viele Optionen.

Aufgabe 2 (Übersetzung ins Relationenmodell) 10 Punkte

Ein Fan von Silvester-Feuerwerkskörpern hat folgendes ER-Schema entworfen:



Man kann Feuerwerksartikel entweder einzeln kaufen, oder als Sortiment. Ein Sortiment besteht aus mehreren einzelnen Feuerwerksartikeln. Es wird auch abgespeichert, welche Produkte (Sortimente oder einzelne Artikel) in welchem Jahr zu welchem Preis gekauft wurden. Für die einzelnen Feuerwerksartikel wird ihre Klasse festgehalten (I: für unter 18 Jahre, II: über 18 Jahre) und möglicherweise eine Schulnote zur Bewertung, wie schön der Effekt war.

Ihre Aufgabe ist nicht, daß ER-Schema zu bewerten, sondern die folgenden Übersetzungen in das relationale Modell. Die relationalen Schemata sind jeweils einmal in der in der Vorlesung verwendeten Kurznotation aufgschrieben, und einmal als `CREATE TABLE` Anweisungen.

Machen Sie jeweils genau ein Kreuz pro Teilaufgabe bei der jeweils besten Antwort.

Entwickler A übersetzt das Schema folgendermaßen ins Relationenmodell:

- Entity-Typ PRODUKT mit seinen Untertypen:

  PRODUKTE(PRODID, NAME, BEMERKUNG°, TYP, KLASSE°, NOTE°)

  Das Attribut "TYP" wird so eingeschränkt, daß es nur die Werte 'A" (für ARTIKEL) und "S" (für SORTIMENT) annehmen kann:

```
CREATE TABLE PRODUKTE(
             PRODID    NUMERIC(4) NOT NULL PRIMARY KEY
                       CHECK(PRODID > 0),
             NAME      VARCHAR(40) NOT NULL,
             BEMERKUNG VARCHAR(1000),
             TYP       CHAR(1) NOT NULL
                       CHECK(TYP IN ('A','S')),
             KLASSE    VARCHAR(5),
             NOTE      NUMRIC(2,1)
                       CHECK(NOTE BETWEEN 0.7 AND 5.0))
```

- Schwacher Entitytyp KAUF mit seinem Relationship:

  KAEUFE(PRODID → PRODUKTE, JAHR, PREIS)

```
CREATE TABLE KAEUFE(
             PRODID NUMERIC(4) NOT NULL PRIMARY KEY
                    REFERENCES PRODUKTE,
             JAHR   NUMERIC(4) NOT NULL
                    CHECK(JAHR BETWEEN 1900 AND 3000)
             PREIS  NUMERIC(5,2) NOT NULL
                    CHECK(PREIS >= 0))
```

- Association Entity-Typ KOMPONENTE mit seinen Relationships:

  KOMPONTEN(ARTIKEL → PRODUKTE, SORTIMENT → PRODUKTE, MENGE)

```
CREATE TABLE KOMPONENTEN(
             ARTIKEL   NUMERIC(4) NOT NULL
                       REFERENCES PRODUKTE,
             SORTIMENT NUMERIC(4) NOT NULL
                       REFERENCES PRODUKTE,
             MENGE     NUMERIC(3) NOT NULL
                       CHECK(MENGE > 0),
             PRIMARY KEY(ARTIKEL, SORTIMENT))
```

a) Betrachten Sie nächst die Tabelle `PRODUKTE` als Übersetzung des Obertyps und seiner Untertypen.

PRODUKTE(<u>PRODID</u>, NAME, BEMERKUNG°, TYP, KLASSE°, NOTE°)

Kreuzen Sie die jeweils beste Möglichkeit an:

- ☐ Man braucht noch folgenden CHECK-Constraint:
  `CHECK(TYPE='A' OR KLASSE IS NULL)`
- ☐ Man braucht noch folgenden CHECK-Constraint:
  `CHECK(TYPE<>'A' OR KLASSE IS NOT NULL)`
- ☐ Man braucht beide Constraints.
- ☐ Beide Constraints sind unnötig.

b) Entsprechend sei das optionale Attribut NOTE von ARTIKEL betrachtet. Kreuzen Sie die jeweils beste Möglichkeit an:

- ☐ Man braucht noch folgenden CHECK-Constraint:
  `CHECK(TYPE='A' OR NOTE IS NULL)`
- ☐ Man braucht noch folgenden CHECK-Constraint:
  `CHECK(TYPE='A' AND NOTE IS NULL)`
- ☐ Man braucht beide Constraints.
- ☐ Beide Constraints sind unnötig.

c) Gibt es bei der Tabelle `KAEUFE` einen Fehler?

KAEUFE(<u>PRODID</u> → PRODUKTE, JAHR, PREIS)

- ☐ Der Schlüssel ist falsch. Man muß `JAHR` zum Schlüssel hinzunehmen.
- ☐ Der Schlüssel ist falsch. Um die Äquivalenz zum ER-Diagramm herzustellen, muß nur `JAHR` alleine Schlüssel sein.
- ☐ Die minimale Kardinalität 1 auf der Seite von KAUF wird durch das relationale Schema nicht garantiert.
- ☐ Die Übersetzung ist korrekt.

d) Gibt es bei der Tabelle `KOMPONENTEN` einen Fehler?

`KOMPONTEN(`<u>`ARTIKEL`</u>` → PRODUKTE, `<u>`SORTIMENT`</u>` → PRODUKTE, MENGE)`

- ☐ Der Schlüssel ist falsch. Man muß in diesem Fall einen künstlichen Schlüssel einführen.
- ☐ Das relationale Schema erlaubt z.B. daß ein Sortiment Komponente eines anderen Sortiments ist. Beim ER-Schema ist das ausgeschlossen.
- ☐ Das Attribut `MENGE` muß Nullwerte erlauben, da nicht jeder Artikel zu jedem Sortiment in Beziehung steht.
- ☐ Keiner der obigen Fehler.

e) Gibt es eventuell ein Problem mit dem Relationship zwischen SORTIMENT und KOMPONENTE?

- ☐ Das relationale Schema garantiert nicht die minimale Kardinalität auf der SORTIMENTS-Seite.
- ☐ Das relationale Schema erlaubt eine viele-zu-viele Beziehung zwischen KOMPONENTEN und SORTIMENTEN. Beim ER-Schema ist diese Beziehung eins-zu-viele.
- ☐ Keine der obigen Aussagen stimmt.

Entwickler B übersetzt das Schema folgendermaßen ins Relationenmodell:

- Entity-Typ PRODUKT mit seinen Untertypen:

```
SORTIMENT(PRODID, NAME, BEMERKUNG°)
ARTIKEL(PRODID, NAME, BEMERKUNG°, KLASSE, NOTE°)

CREATE TABLE SORTIMENTE(
             PRODID    NUMERIC(4) NOT NULL PRIMARY KEY
                       CHECK(PRODID > 0),
             NAME      VARCHAR(40) NOT NULL,
             BEMERKUNG VARCHAR(1000))
CREATE TABLE ARTIKEL(
             PRODID    NUMERIC(4) NOT NULL PRIMARY KEY
                       CHECK(PRODID > 0),
             NAME      VARCHAR(40) NOT NULL,
             BEMERKUNG VARCHAR(1000),
             KLASSE    VARCHAR(5) NOT NULL,
             NOTE      NUMRIC(2,1)
                       CHECK(NOTE BETWEEN 0.7 AND 5.0))
```

- Schwacher Entitytyp `KAUF` mit seinem Relationship:

```
SORTIMENT_KAEUFE(PRODID → SORTIMENTE, JAHR, PREIS)
ARTIKEL_KAEUFE(PRODID → ARTIKEL, JAHR, PREIS)

CREATE TABLE SORTIMENT_KAEUFE( -- ARTIKEL_KAEUFE entsprechend
             PRODID NUMERIC(4) NOT NULL
                    REFERENCES SORTIMENTE,
             JAHR   NUMERIC(4) NOT NULL
                    CHECK(JAHR BETWEEN 1900 AND 3000)
             PREIS  NUMERIC(5,2) NOT NULL CHECK(PREIS >= 0),
             PRIMARY KEY(PRODID, JAHR))
```

- Association Entity-Typ `KOMPONENTE` mit seinen Relationships:

```
KOMPONTEN(VON → SORTIMENTE, IST → ARTIKEL, MENGE)

CREATE TABLE KOMPONENTEN(
             VON   NUMERIC(4) NOT NULL REFERENCES SORTIMENTE,
             IST   NUMERIC(4) NOT NULL REFERENCES ARTIKEL,
             MENGE NUMERIC(3) NOT NULL
                   CHECK(MENGE > 0),
             PRIMARY KEY(VON, IST))
```

f) Betrachten Sie die Übersetzung der Typ-Hierarchie mit zwei Tabellen.

```
SORTIMENT(PRODID, NAME, BEMERKUNG°)
ARTIKEL(PRODID, NAME, BEMERKUNG°, KLASSE, NOTE°)
```

- ☐ Daten werden hier redundant abgespeichert.
- ☐ Es gibt hier keine doppelte Abspeicherung der gleichen Daten.

g) Betrachten Sie die Übersetzung des schwachen Entity-Typs “KAUF”:

```
SORTIMENT_KAEUFE(PRODID → SORTIMENTE, JAHR, PREIS)
ARTIKEL_KAEUFE(PRODID → ARTIKEL, JAHR, PREIS)
```

- ☐ Die Schlüssel stimmen nicht.
- ☐ Daten werden hier redundant abgespeichert.
- ☐ Keines dieser Probleme.

h) Betrachten Sie die Übersetzung des Association-Entity-Typs:

```
KOMPONTEN(VON → SORTIMENTE, IST → ARTIKEL, MENGE)
```

- ☐ Der Schlüssel stimmt nicht. Man muß “VON” als Primärschlüssel und “IST” als Alternativschlüssel deklarieren (oder umgekehrt).
- ☐ Nach dem relationalen Schema kann z.B. ein Sortiment Teil eines Sortiments sein, nach dem ER-Schema nicht.
- ☐ Nach dem relationalen Schema ist es möglich, daß ein Sortiment keine Artikel enthält. Nach dem ER-Schema ist dies ausgeschlossen.
- ☐ Keines dieser Probleme.

i) Designer A kritisiert diesen Vorschlag, weil die gleiche `PRODID` in `SORTIMENTE` und in `ARTIKEL` vorkommen kann, was nach dem ER-Schema nicht möglich ist.

- ☐ Das stimmt nicht.
- ☐ Es stimmt, ist aber nicht wirklich ein Problem, weil durch die Trennung in zwei Tabellen der Typ gewissermaßen Bestandteil des Schlüssels wird (Schlüssel müssen ja nur innerhalb von Tabellen eindeutig sein). Notfalls könnte man bestimmte Intervalle von `PRODID`s für die beiden Untertypen reservieren. Das kann man mit normalen `CHECK`-Constraints überwachen.
- ☐ Es stimmt und ist ein ernstes Problem, das sich nicht mit den Standard-Constraints von SQL lösen läßt. Diese Übersetzung ist hier nicht angemessen.

Schließlich gibt es noch Designer C. Er hat im Prinzip die gleiche Lösung gewählt wie Designer B, nur den Entity-Typ KAUF hat er anders übersetzt:

KAEUFE(<u>PRODID</u> → SORTIMENTE, <u>PRODID</u> → ARTIKEL, <u>JAHR</u>, PREIS)

```
CREATE TABLE KAEUFE(
             SORTID NUMERIC(4) NOT NULL
                    REFERENCES SORTIMENTE,
             ARTID  NUMERIC(4) NOT NULL
                    REFERENCES ARTIKEL,
             JAHR   NUMERIC(4) NOT NULL
                    CHECK(JAHR BETWEEN 1900 AND 3000)
             PREIS  NUMERIC(5,2) NOT NULL CHECK(PREIS >= 0),
             PRIMARY KEY(SORTID, ARTID, JAHR))
```

j) Was würden Sie zu diesem Vorschlag sagen?

- ☐ Der Schlüssel ist falsch. Man muß hier zwei Schlüssel deklarieren: `SORTID` und `ARTID`.
- ☐ Es müßte immer eins der Attribute `SORTID` und `ARTID` null sein, das geht aber nicht, weil sie Bestandteil des Primärschlüssels sind.
- ☐ Es würde so funktionieren und hätte eine Tabelle weniger als der obige Entwurf.

Aufgabe 3 (Relationale Normalformen) 6 Punkte

Kreuzen Sie die jeweils richtige bzw. beste Antwort an. Machen Sie pro Unteraufgabe nur ein Kreuz. Pro Unteraufgabe gibt es einen Punkt. Falls Sie die Lösung nicht wissen, raten Sie (wenn Sie nichts ankreuzen, haben Sie den Punkt auf jeden Fall verloren).

Gegeben sei folgende Tabelle mit einem Telefonverzeichnis:

TELEFONVERZEICHNIS						
TEL	PNR	TYP	NAME	GEBÄUDE	RAUM	INSTITUT
24740	101	P	Stefan Brass	V.Seckendorff P.1	313	Informatik
24771	101	S	Stefan Brass	V.Seckendorff P.1	313	Informatik
24750	102	P	Ramona Vahrenhold	V.Seckendorff P.1	324	Informatik
24771	103	P	Sigrid Kokert	V.Seckendorff P.1	324	Informatik
24737	104	P	Annemarie Herrmann	V.Seckendorff P.1	315	Informatik
24737	105	P	Christian Goldberg	V.Seckendorff P.1	315	Informatik
67890	106	P	Wolfgang Schmidt	V.Seckendorff P.1	123	Physik
10101	107	P	Wolfgang Schmidt	T.-Lieser-Str. 5	456	Mathematik

`PNR` ist die Personalnummer, die die Personen eindeutig identifiziert. Der Name ist nicht unbedingt eindeutig, wie die letzten Zeilen zeigen. Eine Person kann mehrere Telefonnummern haben, wie die ersten beiden Zeilen zeigen. Diese Telefonnummern werden dann über den “`TYP`” unterschieden: “`P`” für die persönliche Telefonnummer, “`S`” für das Sekretariat. Ein Telefon kann zwischen mehreren Personen geteilt werden (z.B. Frau Herrmann und Herr Goldberg).

Gegeben seien folgende funktionale Abhängigkeiten:

- `PNR` → `NAME`, `RAUM`, `GEBÄUDE`
- `TEL` → `RAUM`, `GEBÄUDE`
- `RAUM`, `GEBÄUDE` → `INSTITUT`
- `PNR`, `TYP` → `TEL`
- `TEL`, `PNR` → `TYP`

a) Ein Entwickler schlägt vor, man sollte noch zusätzliche die funktionale Abhängigkeit

`PNR` $\rightarrow$ `TEL`

fordern. Was meinen Sie dazu?

- ☐ Diese funktionale Abhängigkeit ist schon von den gegebenen FAen impliziert. Man braucht sie daher nicht zusätzlich zu fordern.
- ☐ Diese funktionale Abhängigkeit ist schon in den Beispieldaten verletzt. Deswegen kann man sie nicht fordern.
- ☐ Man könnte diese funktionale Abhängigkeit fordern. Dann wären Personen durch ihre Telefonnummer eindeutig identifiziert.

b) Außerdem wird folgende funktionale Abhängigkeit diskutiert:

`NAME`, `GEBÄUDE`, `RAUM` $\rightarrow$ `PNR`

fordern. Was meinen Sie dazu?

- ☐ Das würde Sinn machen. Verletzungen dieser funktionalen Abhängigkeit wären mindestens sehr merkwürdig.
- ☐ Diese funktionale Abhängigkeit ist schon in den Beispieldaten verletzt. Deswegen kann man sie nicht fordern.
- ☐ Diese funktionale Abhängigkeit ist zwar nicht von den Beispieldaten verletzt, aber das ist nur Zufall. Sie dient aber keinem sinnvollen Zweck.

c) Implizieren die gegebenen fünf FAen (d.h. ohne die aus a) und b)) folgende funktionale Abhängigkeit?

`TEL` $\rightarrow$ `INSTITUT`?

Kreuzen Sie die beste Möglichkeit an:

- ☐ Ja.
- ☐ Ja, diese funktionale Abhängigkeit ist sogar trivial.
- ☐ Nein.

d) Was wäre/wären der/die Schlüssel für die Tabelle `TELEFONVERZEICHNIS` bei den gegebenen fünf funktionalen Abhängigkeiten, d.h. ohne die FA aus a) und b)? Genau eine Möglichkeit ist richtig.

- ☐ `TEL`
- ☐ `PNR`
- ☐ Sowohl `TEL` wäre ein Schlüssel, als auch `PNR` wäre ein Schlüssel
- ☐ `TEL` und `PNR` zusammen sind der einzige Schlüssel.
- ☐ `TEL` und `PNR` zusammen sind ein Schlüssel, `PNR` und `TYP` zusammen ein anderer Schlüssel.
- ☐ `TEL`, `PNR`, `TYP` ist der minimale Schlüssel.

e) Ist die Tabelle in BCNF?

- ☐ Ja.
- ☐ Nein.

Falls nein, geben Sie bitte eine formale Begründung an (indem Sie sich auf die Bedingung der Definition beziehen):

____

____

f) Ist die Tabelle in 4NF?

- ☐ Ja.
- ☐ Nein.

Begründen Sie Ihre Antwort bitte ganz kurz:

____

____