Databases Autumn 2025

Hand-In Exercise 1

November 6, 2025

Aiysha Frutiger
Jannick Seper
Luis Tritschler

Total Points	

Task	**Points**

Aiysha Frutiger
Jannick Seper
Luis Tritschler

Task 1

a) `AHV` → `(FullName, Birthday, Zip, country)`
`FullName` → `(FirstName, LastName)`
`Birthday` → `YearOfBirth`
`(County, Zip)` → `City`

The `AHV` uniquely identifies each resident. From these attributes, all others can be derived transitively: `FullName` determines `(FirstName, LastName)`, `Birthday` determines `YearOfBirth`, and `(Country, Zip)` determines `City`. We thought about including `(Country, City)` → `Zip` but there are cities with multiple zips (Zürich for example) and that's why we did not include this dependency.

b) With this functional dependencies we compute the attribute closure for `AHV`. F^+ is `(AHV, FullName, Birthday, Zip, Country, FirstName, LastName, YearOfBirth, City) = sch(Resident)`. Since no subset of `AHV` determines all attributes, `AHV` is minimal and therefore the only candidate key.

c) The relation Resident is in 2NF, since the only candidate key is AHV and therefore no partial dependencies on a subset of a composite key can exist. It is not in Third Normal Form, because there are several transitive dependencies. For example: `AHV` → `(Zip, country)` and `(County, Zip)` → `City`, hence City is transitively dependent on AHV. Version in 3NF:
`Resident(AHV, FullName, Birthday, Zip, Country)`
`Name(FullName, FirstName, LastName)`
`Birthday(Birthday, YearOfBirth)`
`Location(Zip, Country, City)`

Task 2

a) `{{A, B, C}, {A, C, D}}`

b)
 i Because for FD2 `B` on the left is not a superkey, R is not in BCNF.
 ii Because for FD3 `E` on the right which is not part of a primary key, R is not in 3NF.
 iii Because in FD3, `E` (non-prime attribute) is dependent on `C,D` which is a subset of a candidate key, R is not in 2NF

 R is only in 1NF.

Task 3

a) `{{A, C}, {A, B}}`

b)
 i S is in 2NF because in FD1, neither `A` nor `B` alone determine `D` and `C` is in the candidate keys. Also in FD2, although `C` is a proper subset of a candidate key, B is a primary key and therefore all rules for 2NF hold true.

ii S is also in 2NF because in FD1, `A,B` is a candidate key and therefore a superkey and therefore 2NF is satisfied. Because `C` alone is not a superkey, but `B` is a prime key, this FD also satisfies 3NF.

iii However, because `C` alone is not a superkey, it does not satisfie FD2 and therefore is not a BCNF.

Task 4

Task 5

a) **Task 5(a) – Candidate Keys of Relation U Given:** Functional dependencies:

$$\{\, \mathtt{CU} \rightarrow \mathtt{D},\ \mathtt{D} \rightarrow \mathtt{E,V,W,U,A},\ \mathtt{U} \rightarrow \mathtt{B},\ \mathtt{V} \rightarrow \mathtt{A} \,\}.$$

Identify attributes that must be in every key No functional dependency determines `C`, therefore `C` must be included in every candidate key.

Determine attribute closures

- For `{C,U}`: From `CU → D` we obtain `D`. Then, `D → E,V,W,U,A` and `U → B`. Hence,

$$(\mathtt{CU})^+ = \{\mathtt{A,B,C,D,E,U,V,W}\}.$$

  ⇒ `{C,U}` is a superkey. Since neither $\mathtt{C}^+$ nor $\mathtt{U}^+$ contains all attributes, `{C,U}` is **minimal** and therefore a **candidate key**.

- For `{C,D}`: From `D → U,A,E,V,W` and `U → B` we obtain

$$(\mathtt{CD})^+ = \{\mathtt{A,B,C,D,E,U,V,W}\}.$$

  ⇒ `{C,D}` is a superkey. Since neither `C` nor `D` alone determines all attributes, it is also **minimal** and thus a **candidate key**.

Check other combinations Other sets such as `{C,V}` or `{C,A}` do not determine `U` or `D`, and therefore cannot be superkeys. Supersets of `{C,U}` or `{C,D}` are not minimal.

Result:

$$\boxed{\text{The candidate keys of } \mathtt{U} \text{ are } \{\mathtt{C,U}\} \text{ and } \{\mathtt{C,D}\}.}$$

b) **Given:** Relation `U(A,B,C,D,E,U,V,W)` with functional dependencies

$$\{\, \mathtt{CU} \rightarrow \mathtt{D},\ \mathtt{D} \rightarrow \mathtt{E,V,W,U,A},\ \mathtt{U} \rightarrow \mathtt{B},\ \mathtt{V} \rightarrow \mathtt{A} \,\}.$$

Check 1NF All attribute domains are atomic.

$$\Rightarrow \mathtt{U} \text{ is in 1NF.}$$

Check 2NF A relation is in 2NF if every non-prime attribute is fully functionally dependent on each candidate key. Candidate keys are `{C,U}` and `{C,D}`.

- `U → B`: `U` is only part of key `{C,U}` ⇒ partial dependency.
- `D → E,V,W`: `D` is only part of key `{C,D}` ⇒ partial dependency.

⇒ *Violates 2NF.*

Check 3NF A relation is in 3NF if, for every FD `X → A`, either `X` is a superkey or `A` is a key attribute. Since `D → E,V,W` and `U → B` hold, and neither `D` nor `U` are superkeys, these dependencies violate 3NF.

⇒ `U` is not in 3NF.

Check BCNF In BCNF, every determinant must be a superkey. Dependencies `D → E,V,W`, `U → B`, and `V → A` violate this rule.

⇒ `U` is not in BCNF.

Result: Since `U` violates 2NF (and therefore 3NF and BCNF), it is only in:

$$\boxed{\text{First Normal Form (1NF).}}$$

c) **Minimal Cover (Canonical Form)**

Given functional dependencies (already minimal on both sides):

$$\{\, \mathtt{CU} \to \mathtt{D},\ \mathtt{D} \to \mathtt{E,V,W,U,A},\ \mathtt{U} \to \mathtt{B},\ \mathtt{V} \to \mathtt{A} \,\}.$$

No attribute on the left–hand side can be removed (neither `C` nor `U` in `CU→D`), and none of the dependencies is implied by the others. Hence, this is already the **minimal cover**.

3NF Synthesis

For each determinant, create one relation that includes all attributes from its FD group. Since one of them (R_{CU}) contains the candidate key `{C,U}`, the resulting decomposition will be lossless and dependency-preserving.

$$\begin{aligned}
R_{CU}(\mathtt{C,U,D}) &\quad \text{from } \mathtt{CU{\to}D}\\
R_{D}(\mathtt{D,U,A,E,V,W}) &\quad \text{from } \mathtt{D{\to}U,A,E,V,W}\\
R_{U}(\mathtt{U,B}) &\quad \text{from } \mathtt{U{\to}B}\\
R_{V}(\mathtt{V,A}) &\quad \text{from } \mathtt{V{\to}A}
\end{aligned}$$

Since R_{CU} includes the key `{C,U}`, the natural join of all projections is **lossless**, and all dependencies are **preserved** (property of the 3NF-synthesis algorithm).

Normal Forms of the Resulting Relations

- R_{CU}(`C,U,D`) – Contains FDs `CU→D` and `D→U`. – Satisfies 3NF (each non-trivial FD has either a key or a key attribute on the RHS). – Not in BCNF because `D→U` holds while `D` is not a key.

Aiysha Frutiger
Jannick Seper
Luis Tritschler

- $\mathtt{R_D(D,U,A,E,V,W)}$ – `D` is a key for all its dependencies → BCNF. – However, `V→A` also holds globally and both attributes occur here, so to maintain BCNF we separate $\mathtt{R_V(V,A)}$.
- $\mathtt{R_U(U,B)}$ – `U` is key → BCNF.
- $\mathtt{R_V(V,A)}$ – `V` is key → BCNF.

Variant B (cleaner BCNF form): Split $\mathtt{R_D}$ into individual relations for each RHS attribute:

$$\mathtt{R_{DU}(D,U)},\ \mathtt{R_{DA}(D,A)},\ \mathtt{R_{DE}(D,E)},\ \mathtt{R_{DV}(D,V)},\ \mathtt{R_{DW}(D,W)}.$$

Each of these is in BCNF because `D` is the key in its relation. $\mathtt{R_{CU}(C,U,D)}$ remains in 3NF (not BCNF). $\mathtt{R_U(U,B)}$ and $\mathtt{R_V(V,A)}$ are BCNF.

Final Normal Forms

$$\begin{aligned}\mathtt{R_{CU}(C,U,D)} &\to \text{3NF (not BCNF)}\\ \mathtt{R_{DU}(D,U)},\ \mathtt{R_{DA}(D,A)},\ \mathtt{R_{DE}(D,E)},\ \mathtt{R_{DV}(D,V)},\ \mathtt{R_{DW}(D,W)} &\to \text{BCNF}\\ \mathtt{R_U(U,B)},\ \mathtt{R_V(V,A)} &\to \text{BCNF}\end{aligned}$$

Result: The decomposition is **lossless** (a key relation is retained) and **dependency-preserving**. All relations are in BCNF except $\mathtt{R_{CU}}$, which remains in 3NF to ensure dependency preservation.

$$\boxed{\text{Highest possible normal form with dependency preservation: 3NF.}}$$

d) **Goal:** Find a lossless BCNF decomposition of `U` and compare it to the 3NF decomposition from (c).

Observation. In the 3NF design of (c), the only non-BCNF relation is $\mathtt{R_{CU}(C,U,D)}$ due to the dependency `D → U` with `D` not being a key of $\mathtt{R_{CU}}$.

BCNF Step. Decompose $\mathtt{R_{CU}(C,U,D)}$ using `D → U`:

$$\mathtt{R_{CU}(C,U,D)} \ \Rightarrow\ \mathtt{R_{CD}(C,D)} \quad \text{and} \quad \mathtt{R_{DU}(D,U)}.$$

This split is **lossless** because the common attribute `D` is a key in $\mathtt{R_{DU}}$.

Final BCNF schema. Starting from the 3NF set in (c) and replacing $\mathtt{R_{CU}}$ by its BCNF split, we obtain:

$$\mathtt{R_{CD}(C,D)},\ \mathtt{R_{DU}(D,U)},\ \mathtt{R_{DA}(D,A)},\ \mathtt{R_{DE}(D,E)},\ \mathtt{R_{DV}(D,V)},\ \mathtt{R_{DW}(D,W)},$$
$$\mathtt{R_U(U,B)},\ \mathtt{R_V(V,\ A)}.$$

All these relations are in **BCNF**:

- Each $\mathtt{R_{D*}}$ has determinant `D` which is a key in that relation.
- $\mathtt{R_U(U,B)}$ and $\mathtt{R_V(V,A)}$ have keys `U` and `V`, respectively.
- $\mathtt{R_{CD}(C,D)}$ contains no nontrivial projected FD with a non-key LHS, hence BCNF.

Aiysha Frutiger
Jannick Seper
Luis Tritschler

Losslessness. Each BCNF split is lossless (decomposition by a dependency `X→Y` is lossless when the common part contains a key of one component). Since we only refined R_{CU} by such a split, the overall join remains **lossless**.

Dependency preservation and comparison to (c).

- In (c), the 3NF design preserves all original dependencies, notably `CU → D` in R_{CU}.
- In the BCNF design above, `CU → D` is *not* preserved in any single relation (it can only be checked via a join), while the other dependencies (`D → U,A,E,V,W`, `U → B`, `V → A`) remain preserved in their respective relations.

Answer. The BCNF decomposition differs from (c) by replacing R_{CU}`(C,U,D)` with R_{CD}`(C,D)` and R_{DU}`(D,U)`. It is **lossless** but not fully **dependency-preserving** (the FD `CU → D` is lost as a directly enforceable dependency). All resulting relations are in **BCNF**.