Database Systems Assignment

Arnav Kumar Behera

This folder contains the source code for

- **Lossless Join Testing**
- **Lossless and Dependency Preserving Decomposition into 3NF**
- **Lossless Decomposition into BCNF**

Features

- Written in C++
- Command Line Interface
- Easy on the eyes and simple to use.

Prerequisites

- **gcc 11.3.0**
- **C++20 standard**

How to get it running?

Change working directory into the directory that contains the folde

- **Way 1:**

  The three codes are in C++ and can be compiled and executed using g++.

  ```
  #-------------------------------------
  # filename can be
  # ljt.cpp(for lossless join testing)
  # 3nf.cpp (for decomposition into 3nf)
  # bcnf.cpp(for lossless decomposition )
  #-------------------------------------
  g++ <filename> -o <executable filename>
  ./<executable filename>
  ```

- **Way2 (Easier and Recommended):**

  ```
  # Run this in terminal before executing any commands
  source make.sh

  # To run this in command line you can use the run command
  # usage: run filename(without cpp extension) see usage below VVV

  run bcnf            # similarly you can use run 3nf or run ljt

  # To run this from file input(input.txt) and write to file(output.txt)
  # usage: run_file filename(without cpp extension) see usage below VVV

  run_file bcnf       # similarly you can use run_file 3nf or run_file ljt
  ```

Input Format

Lossless Join Testing :

- First line is n (an integer denoting the number of attributes in base table)
- These attributes will be named A,B,C.. so write the FDs keeping those in mind
- Second line is m (an integer denoting the number of attributes in base table)
- The next m lines are the functional dependencies of the base table (in format A>BC no spaces)
- This line is q (an integer denoting the number of derived tables)
- The next q lines represent the attributes in the derived tables (no spaces)

Sample Input 1:

```
5
4
E>D
D>E
C>AD
AB>C
2
ABC
CDE
```

Output 1: `The above decomposition does not satisfy lossless join property`

Sample Input 2:

```
5
4
E>D
D>E
C>AD
AB>C
2
ABC
CDE
```

Output 2: `The above decomposition satisfies lossless join property`

Lossless and Dependency Preserving Decomposition into 3NF:

- First line is n (an integer denoting the number of attributes in base table)
- Second line is s (a string denoting the attributes in base table (no spaces))
- This line is q (an integer denoting the number of derived tables)
- The next q lines are the functional dependencies of the base table (in format A>BC no spaces)

Sample Input:

```
7
ABCDEFG
3
A>CDE
B>FG
AB>CDEFG
```

Sample Output:

```
R1(ACDE)
R2(BFG)
R3(AB)
```

Lossless Decomposition into BCNF:

- First line is n (an integer denoting the number of attributes in base table)
- Second line is s (a string denoting the attributes in base table (no spaces))
- This line is q (an integer denoting the number of derived tables)
- The next q lines are the functional dependencies of the base table (in format A>BC no spaces)

Sample Input 1:

```
6
ABCDEF
4
A>BC
CD>E
B>D
E>A
```

Output 1:

```
R1(ABC)
R2(ADE)
R3(AF)
```

Sample Input 2:

```
6
ABCDEF
5
A>BC
CD>E
B>D
E>A
A>F
```

Output 2:

```
R1(BD)
R2(ABCEF)
```

Have fun trying it out.

License

MIT

Contact

20CS01070@iitbbs.ac.in