



Institut für Informatik - Lehrstuhl Datenbanken

Datenbanken I

Wintersemester 09/10 - Dr. Alexander Hinneburg

Übung 1: Datenverarbeitung mit Dateien (Abgabe bis 14.10.2009, 23.59 Uhr) 10 Punkte

Eine Anwendung, die uns in dieser Vorlesung begleiten wird, beschreibt Studenten und Veranstaltungen (in der Regel Vorlesungen, Seminare und Übungen). Es wird erfaßt, welcher Student welche Veranstaltung besucht. Ein Student kann mehrere Veranstaltungen besuchen. Ebenso besuchen eine Veranstaltung in der Regel mehrere Studenten. Die Daten werden durch drei CSV-Dateien¹ `student.csv`, `veranstaltung.csv` und `besuch.csv` beschrieben. Die CSV-Dateien sind in UTF-8 kodiert. Eine Zeile wird durch Kommata in Felder aufgeteilt, Stringfelder werden in Anführungszeichen eingeschlossen². Die Dateien haben die folgenden Felder in der angegebenen Reihenfolge:

student.csv SID, Vorname und Nachname

veranstaltung.csv VID, Veranstaltungstitel

besuch.csv VID und SID

Sie können mit diesen Dateien zum Beispiel herausfinden, welche Veranstaltungen die Studentin Mia Kapp besucht. Dazu müssen Sie die SID von Mia Kapp in `student.csv` finden. Dann suchen Sie aus `besuch.csv` heraus, welche VIDs in der gleichen Zeile wie die SID von Mia Kapp stehen. Wenn Mia Kapp mehrere Veranstaltungen besucht, gibt es in `besuch.csv` auch mehrere Zeilen mit seiner SID. Zum Schluß suchen Sie zu jeder gefundenen VID den Titel der Veranstaltung aus `veranstaltung.csv` heraus und geben den Titel aus.

Hinweise

- Sie dürfen zum Schreiben der Programme eine beliebige Programmiersprache nutzen, z.B. C#, C++, Java, Perl oder PHP.
- Sie dürfen gern Bibliotheken zum Einlesen von CSV-Dateien verwenden. Die eigentlichen Aufgaben dürfen aber nicht an Bibliotheksfunktionen delegiert werden. Die Programme müssen auf `anubis.informatik.uni-halle.de` in Ihrem Home-Verzeichnis von der Kommandozeile aus kompilierbar oder interpretierbar sein. Die Ausgaben sollen ausschließlich auf die Kommandozeile gehen, keine Fenster aufmachen.

¹Zip-Datei `csv_dateien.zip` ist im StudIP → DB I → Dateien → Material.

²Ausführliche Beschreibung des CSV-Formats <http://www.rfc-editor.org/rfc/rfc4180.txt>

- Sie müssen die Programme in der Übung am Rechner vorführen können und in der Lage sein, kleine Änderungen an der Eingabe vorzunehmen (z.B. statt Mia Kapp, Edwin Quicker).
- Die Eingaben der Programme sollen direkt in den Quellcode reingeschrieben werden. Basteln Sie keine Eingaben über Kommandozeile oder ähnliches. Es ist in Ordnung, wenn Sie für eine neue Eingabe, das Programm neu kompilieren müssen.
- Kommentieren Sie Ihr Programm auf verständliche Weise. Programme ohne Kommentare werden nicht gewertet.
- Wenn Sie zuviel Zeit zum Schreiben von funktionsfähigen Programmen brauchen (> 2h), geben Sie Pseudo-Kode ab. Dadurch bekommen Sie nicht die volle Punktzahl, aber es wird auch gewertet.

Aufgabe 1.1:

4 Punkte

Schreiben Sie ein Programm, das zu einem gegebenen Vornamen alle Studenten mit `SID`, Vornamen und Nachnamen ausgibt. Testen Sie Ihr Programm mit dem Vornamen Mia. Welche Studenten heißen Mia mit Vornamen?

Wieviel CPU-Zeit braucht Ihr Programm auf anubis zur Ausführung? Das können Sie mit dem Kommando `time` messen.

Wieviel Codezeilen hat Ihr Programm? Wie lange haben Sie für die Programmierarbeit gebraucht? Hier soll nur die Programmierarbeit an dem Problem gemessen werden, nicht die Zeit, die Sie brauchen um sich mit den Funktionen des Compilers oder ähnliches vertraut zu machen.

Aufgabe 1.2:

6 Punkte

Schreiben Sie ein Programm, das zu einem gegebenen Vor- und Nachnamen eines Studenten die Titel aller Veranstaltungen ausgibt, welche von dem Studenten besucht werden. Gehen Sie davon aus, dass es nur einen Studenten mit der Kombination Vor- und Nachname gibt. Testen Sie Ihr Programm mit Mia Kapp. Welche Veranstaltungen besucht Mia Kapp?

Wieviel CPU-Zeit braucht Ihr Programm auf anubis zur Ausführung?

Wieviel Codezeilen hat Ihr Programm? Wie lange haben Sie für die Programmierarbeit gebraucht?