

LB01: Ein-/Ausgabeumleitung & stdin/stdout/stderr

Thema / Zweck der Aufgabe

- Wiederholung aus MINT-Labor : bash Kommandos (siehe
- Ein-/Ausgabeumleitung & Pipes in der bash
- C-Programmierung stdin/stdout/stderr

Hilfestellungen / Quellen:

- Wolf, J. – Shell Programmierung; Einführung, Rheinwerk Computing
 - [1.7 Crashkurs: einfacher Umgang mit der Kommandozeile](#)
 - [1.10 Datenstrom](#)
 - 1.10.1 [Ausgabe umleiten](#)
 - 1.10.2 [Standardfehlerausgabe umleiten](#)
 - 1.10.3 [Eingabe umleiten](#)
 - 1.10.4 [Pipes](#)
 - 1.10.5 [Ein T-Stück mit tee](#)

Aufgabenstellungen

1. **Wiederholung** aus SWE bzw. dem Mint-Labor (Einzelaufgabe):

- a) Stellen Sie sicher, dass Sie für das BTS-Labor ein Linux System zur Verfügung haben (Raspbian auf einem Raspberry Pi, Linux Mint,...). Installieren Sie nötigenfalls ein Linux auf Ihrem PC/Notebook (z.B. in einer VM oder WSL,...).
- b) Loggen Sie sich in Ihrem Linux System ein und öffnen Sie ein Terminal-Fenster. Wiederholen Sie die wesentlichsten Terminalbefehle, indem Sie die „**BTS LB 01 - Übungen zu bash Kommandos**“ Schritt-für-Schritt durchmachen (keine schriftliche Abgabe hierfür notwendig!)
- c) Geben Sie eine Liste aller Dateien und Verzeichnisse des Verzeichnisses **/dev** aus. Verwenden Sie hierfür den Parameter **-l** und versuchen Sie, die ausgegebenen Informationen zu verstehen (z.B. die Dateigröße, Zugriffsrechte, usw.)
- d) Laden Sie die Datei <https://www.users.fh-salzburg.ac.at/~rgraf/downloads/70salbumcharts.txt> mittels **wget** auf Ihren Computer. Ändern Sie danach den Namen der Datei auf **fhxxxxx_a70charts.txt** (der Teil **fhxxxxx** im Dateinamen steht für Ihre FHS-Benutzerkennung)
- e) Geben Sie nun die Anzahl der Textzeilen und etwaige andere Dateiinformationen der txt-Datei auf Console aus. (Hinweis: **wc**).

- f) Geben Sie durch die Verkettung mehrerer Kommandos die Zeile 10 bis 15 der obigen txt-Datei aus. Die ausgegebenen Zeilen sollten dabei durchnummeriert werden.
(Hinweis: **head**, **tail**, **cat**)

Hilfestellung: Wenn Sie mal die Parameter eines Linux Konsolenkommandos nicht wissen, dann können Sie sich im Terminalfenster mit **man** weiterhelfen lassen. Das Programm **man** zeigt zu dem meisten Commands eine Hilfeseite an. Versuchen Sie zum Beispiel in der Konsole mal das Kommando **man ls**
Wird Ihnen dabei keine Hilfeseite angezeigt (z.B. wie bei **man cd**), dann wird es sich wahrscheinlich um ein Builtin Command der Shell handeln (siehe **type cd**). Dann können Sie ein **help** versuchen; z.B. erhalten Sie für das Builtin Command **cd** mit **help cd** eine Hilfe.

2. C-Programmierung (in 2-er Teams)

- a) Schreiben Sie ein C-Konsolenprogramm **cvb**, das die in obiger Aufgabe 1.f) beschriebene Funktionalität per Programm anbietet und ausgewählte Zeilen einer Textdatei auf *stdout* ausgibt.

Der Aufruf sollte wie folgt erfolgen können:

cvb [OPTION]... [DATEINAME]

Wenn ein DATEINAME angegeben wird, dann wird die angegebene Datei gelesen. Wenn kein Dateiname oder alternativ als Platzhalter ein Zeichen "-" angegeben wird, dann wird der Eingabestrom aus *stdin* gelesen.

--help	Hilfeinformation wird ausgegeben; Programm terminiert
-s n	Angabe der ersten auszugebenen (n-te) Zeile; wenn keine Angabe, dann wird ab erster Zeile gelesen
-e n	Angabe der letzten auszugebenen (m-te) Zeile; wenn keine Angabe, dann wird bis zur letzten Zeile gelesen
-n [format]	(optional) Ausgabe mit führenden Zeilennummern; optionale Formatparameter hierfür: <i>uint</i> Breite des Zeilennummernfelds (numerisch) R rechtsbündig (default) O rechtsbündig mit führenden Nullen L linksbündig N die Zeilennummer beginnt mit Null Beispiel: 04 ... Ausgabe der Zeilennummer rechtsbündig mit führenden Nullen, z.B. 0017
-v	Programm gibt während der Prozessierung etwaige zusätzliche Status- und Dateiinformationen aus
-q	Programm gibt nur etwaige zusätzliche Status- und Dateiinformationen aus, ohne jedoch den Inhalt der Datei auszugeben
--version	Versionsinformation des Programms wird ausgegeben

Achtung: Fehler dürfen in unserem Programm grundsätzlich nur auf *stderr* ausgegeben werden!

Aufrufbeispiele:

cvb -s 8 -e 12 fhsxxxxx_devfiles.txt

- gibt die Zeile 8 bis 12 der Datei fhsxxxxx_devfiles.txt aus

cvb -s 8 -c 12 fhsxxxxx_devfiles.txt

- gibt die Zeile 8 bis 12 der Datei fhsxxxxx_devfiles.txt aus

cat fhsxxxxx_devfiles.txt | cvb -b 15 -n

- liest die Daten aus der Pipe (*stdin*) von Beginn an und gibt diese bis Zeile 15 aus; die Zeilen werden mit vorangestellten Zeilennummern ausgegeben

- b) Verwenden Sie Ihr Programm und geben Sie von der Datei [70salbumcharts.txt](https://www.users.fh-salzburg.ac.at/~rgraf/downloads/70salbumcharts.txt)¹ die Zeile 8 bis 18 auf *stdout* aus. Stellen Sie beim Aufruf des Programms sicher, dass etwaige Fehlermeldungen durch eine entsprechende Ausgabeumleitung unterdrückt werden.

Abgabe:

Form der Abgabe

- **Quelltexte der Aufgabe 2**
- **readme Datei** mit einer kurzen Erklärung des Programms, wie das Programm verwendet und wie das Programm kompiliert werden kann
- **Screenshots** des Terminalfensters in eine **PDF-Datei** für
 - Aufgabe 1.d und 1.f
 - Aufgabe 2.a mit mehreren (!!) **Aufrufbeispielen** (siehe obige Aufrufbeispiele)
- **Abgabe: PDF-Datei** und Quelltext in eine gemeinsame Datei gezippt
Name der zip-Datei: **BTS_LB01_Nachname1.Nachname2.zip**

Einzelabgabe in **Moodle vor der nächsten Laborstunde**

¹ <https://www.users.fh-salzburg.ac.at/~rgraf/downloads/70salbumcharts.txt>