

Aufgaben zur Klausur **Unix** im WS 2012/13 (IA 351)

Zeit: 75 Minuten

erlaubte Hilfsmittel: keine

Bitte tragen Sie Ihre Antworten und fertigen Lösungen ausschließlich an den gekennzeichneten Stellen in das Aufgabenblatt ein. Ist ihre Lösung wesentlich umfangreicher, so überprüfen Sie bitte nochmals Ihren Lösungsweg.

Nutzen Sie die Rückseiten der Klausur zur Entwicklung der Lösungen und übertragen die fertigen Lösungen in das Aufgabenblatt.

Sollten Unklarheiten oder Mehrdeutigkeiten bei der Aufgabenstellung auftreten, so notieren Sie bitte, wie Sie die Aufgabe interpretiert haben.

Viel Erfolg!

Diese Klausur besteht einschließlich dieses Deckblattes aus 6 Seiten.

Aufgabe 1:

Ein UNIX-Prozeß besitzt im Betriebssystem 3 Datensegment, das Textsegment, das Benutzerdatensegment und das Systemdatensegment.

Was wird bei der Ausführung eines Programms im Textsegment gespeichert?

1)

2)

Was wird bei der Ausführung eines Programms im Benutzerdatensegment gespeichert?

1)

2)

3)

4)

Was wird bei der Ausführung eines Programms im Systemdatensegment gespeichert?

1)

2)

3)

4)

Welche Segmente sind im Benutzermodus schreibgeschützt?

.....

Welche Segmente sind im privilegierten Systemmodus schreibgeschützt?

.....

Was passiert mit den drei Segmenten bei einem Aufruf von `fork`.

.....

.....

.....

.....

.....

Aufgabe 2:

Geben Sie Vorteile von virtueller Adressierung gegenüber realer Adressierung in präzisen Stichworten (bitte keine Allgemeinplätze und Phrasen) an.

1)

2)

3)

4)

5)

Aufgabe 3:

Was versteht man in UNIX unter einem Filter?

.....
.....
.....
.....

Welche Aufrufkonventionen sollte ein `bash`-Skript einhalten, damit es als Filter eingesetzt werden kann?

- 1)
- 2)
- 3)
- 4)

Ist das Kommando `find` ein Filter?

ja nein

Begründung:

.....
.....

Ist das Kommando `bash` ein Filter?

ja nein

Begründung:

.....
.....

Ist das Kommando `ls` ein Filter?

ja nein

Begründung:

.....
.....

Ist das Kommando `xargs` ein Filter?

ja nein

Begründung:

.....
.....

Aufgabe 4:

Entwickeln Sie ein Kommando, das das gesamte Dateisystem nach Dateien durchsucht, die mit `.jpg` oder `.png` enden. Die Liste der Dateinamen soll in einer Datei `bilder` im momentanen Arbeitsverzeichnis gespeichert werden. Mögliche Fehlermeldungen, die beim Suchen auftreten können, sollen nicht angezeigt werden. Während der Suche soll normal weitergearbeitet werden können.

.....
.....

Aufgabe 5:

Gegeben sei eine Datei mit Wörtern für ein Wörterbuch. Die Datei sei so organisiert, dass auf jeder Zeile genau ein Wort steht. Alle Wörter bestehen dabei nur aus den 26 Kleinbuchstaben des ASCII Zeichensatzes. Man findet solche Dateien zum Beispiel auf manchen Unix-Systemen unter `/usr/share/dict/words`.

Geben sie in den folgenden Aufgaben einen regulären Ausdruck an, mit dem man die jeweils gesuchten Wörter mit dem `egrep` Kommando selektieren kann. `egrep` verarbeitet den vollen Umfang der Syntax für reguläre Ausdrücke, `grep` akzeptiert auf manchen Systemen nur eine eingeschränkte Syntax. Beachten Sie bitte, dass `egrep` im Standardverhalten nur nach Teilzeichenreihen sucht und alle Zeilen mit einer gefundenen Teilzeichenreihe ausgibt.

Nutzen Sie nur die in der Vorlesung verwendete Syntax für reguläre Ausdrücke, und keine Erweiterungen, die dort nicht behandelt worden sind.

1. Gesucht sind alle Wörter, die als 2. Buchstaben ein `a` haben und mit `xyz` enden.

.....

2. Gesucht sind alle Wörter, die mindestens drei mal das Zeichen `t` enthalten.

.....

3. Gesucht sind alle Wörter mit vier bis sechs Zeichen

.....

4. Gesucht sind alle Wörter, die genau zwei mal das Zeichen `a` enthalten.

.....

5. Gesucht sind alle Wörter, die ein Doppel-`t` oder ein Doppel-`s` enthalten.

.....

6. Gesucht sind alle Wörter, die mindestens ein `a` und ein `o` enthalten.

.....

7. Gesucht sind alle Wörter, die genau ein `a` und ein `o` enthalten, wobei das `a` als erstes im Wort vorkommen soll.

.....

8. Gesucht sind alle Wörter, die genau ein `a` und ein `o` enthalten, wobei die Reihenfolge nicht festgelegt ist.

.....