
Aufgaben zur Klausur **Unix** im SS 2001 (IA351)

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.

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 8 Seiten

Aufgabe 1:

Geben Sie ein `bash`-Kommando an, mit dem man alle HTML-Dateien im momentanen Arbeitsverzeichnis editiert. Es sollen in jeder Datei alle `<h1>` und `<h2>` tags durch `<H1>` und `<H2>` ersetzt werden. Der alte Dateiinhalt soll in einer Backupdatei erhalten bleiben. Diese soll die zusätzliche Endung `.bak` besitzen.

.....

.....

.....

.....

.....

.....



Aufgabe 2:

Reguläre Ausdrücke sind ein gutes Werkzeug zur Beschreibung von Textmustern. Für den Aufbau von regulären Ausdrücken gibt es folgende Regeln:

1. jedes Zeichen aus dem Alphabet (z.B. dem ASCII Zeichensatz) ist ein regulärer Ausdruck.
2. Sonderzeichen, die zum Aufbau der Ausdrücke verwendet werden, z.B. `\ [] . * + () | ?`, müssen mit einem `\` maskiert werden. `\.` steht also für das Zeichen `.`
3. `.` steht für ein beliebiges Zeichen
4. `[z1z2z3]` steht für eine Menge von Zeichen z_1, z_2, z_3 , hier dürfen auch Intervalle angegeben werden: `[z1-zn]`.
5. `[^z1z2z3]` steht für die Menge aller Zeichen außer z_1, z_2 und z_3 .
6. Wenn r_1 und r_2 reguläre Ausdrücke sind, dann auch r_1r_2 (Folge, Sequenz, r_1 gefolgt von r_2)
7. Wenn r_1 und r_2 reguläre Ausdrücke sind, dann auch $r_1|r_2$ (Alternativen, Auswahl, r_1 oder r_2)
8. r_1^* steht für die 0,1,2,...-fache Wiederholung von r_1
9. r_1^+ steht für die 1,2,...-fache Wiederholung von r_1
10. $r_1?$ steht für r_1 oder die leere Zeichenfolge.
11. (r_1) steht für r_1 , Klammern stehen also zum Zusammenfassen von regulären Ausdrücken.

Beispiele:

`[a-zA-Z0-9]`

steht für die Menge der Buchstaben und Ziffern.

`[^&]`

steht für die Menge aller Zeichen außer `&`.

`[^0-9]`

alles außer Ziffern.

`ab*`

Eine Zeichenfolge, die mit `a` als 1. Zeichen gefolgt von beliebig vielen `b`'s.

`ab+`

Eine Zeichenfolge, die mit `a` als 1. Zeichen gefolgt von beliebig vielen `b`'s aber mindestens einem `b`.

`(ab)*`

Eine Zeichenfolge, die abwechselnd aus `a`'s und `b`'s besteht.

`abc|def`

entweder `abcef` oder `abdef`.

`(abc)|(def)`

entweder `abc` oder `def`.

`a?`

ein `a` oder nichts.

Aufgabe:

Beschreibung von Adressen von WWW-Dokumenten.

Ein Adresse eines WWW-Dokuments besteht aus einer manchmal etwas kryptischen Zeichenfolge, z.B.

`//www.fh-wedel.de:80/~si/index.html?NAME=wert&F00=bar#abc`

Diese Adressen enthalten als erstes einen Rechnernamen eingeleitet durch `//`. Der vollständige Rechnername ist eine Folge von Namen getrennt durch einen `.`. Der Rechnername enthält Buchstaben, Ziffern und `-`.

Beschreiben Sie diesen Aufbau durch einen regulären Ausdruck:

.....

Nach dem Rechnernamen kann optional eine Portnummer folgen, die durch einen `:` eingeleitet wird.

Geben Sie einen regulären Ausdruck für eine optionale Portnummer an:

.....

Danach folgt ein Dokumentenname, dieser besteht aus einem Pfad ähnlich dem eines absoluten Pfadnames im Filesystem, darf aber keinen `#` und keine `?` enthalten.

Beschreiben Sie diesen Aufbau durch einen regulären Ausdruck:

.....

Optional kann noch ein Parameterstring kommen, dieser beginnt mit einem `?` und besteht aus einer Folge von Parametern, die durch `&` getrennt sind. Der Parameterstring darf keinen `#` enthalten.

Ein regulärer Ausdruck für den optionalen Parameterstring:

.....

Als letzter Bestandteil darf eine Marke für eine Position in einem Dokument folgen. Diese wird durch ein `#` eingeleitet und darf beliebige Zeichen enthalten.

Ein regulärer Ausdruck für eine optionale Marke:

.....

Aufgabe 3:

Welche 4 UNIX Systemaufrufe werden zum dynamischen Kreieren, zum Synchronisieren und zum Beenden von Prozessen und zum Starten von Programmen verwendet. Welche Operationen werden von den einzelnen Aufrufen ausgeführt ?

1.
.....
.....
2.
.....
.....
3.
.....
.....
4.
.....
.....

Aufgabe 4:

Welche der folgenden Aussagen treffen zu?

1. *rcs* ist ein UNIX-Kommando zum Kopieren, Löschen und Sichern von Dateien (remove,copy,save). ja nein weiß nicht
 2. Eine effiziente Versionsverwaltung legt für jede Modifikation einer Datei eine vollständige Kopie in einem Archiv an. ja nein weiß nicht
 3. In einem *RCS/SCCS*-Archiv werden mehrere Quelldateien aufbewahrt. ja nein weiß nicht
 4. In einem *CVS*-Archiv werden ganze Verzeichnisbäume abgespeichert. ja nein weiß nicht
 5. *CVS* ist eine Erweiterung von *RCS* ja nein weiß nicht
 6. Versionsverwaltungen, die mit Rückwärts-Deltas arbeiten, sind effizienter als Versionsverwaltungen mit Vorwärts-Deltas. ja nein weiß nicht
 7. Versionsverwaltungssysteme werden bei Softwarefirmen nur sporadisch eingesetzt, da ihre Anschaffung kostspielig ist. ja nein weiß nicht
 8. Versionsverwaltungssysteme haben den Nachteil, daß dadurch die *makefiles* unhandlich werden. ja nein weiß nicht
 9. Versionsverwaltungssysteme werden bei Softwarefirmen nur sporadisch eingesetzt, da sie zu viel Rechenzeit für das Archivieren und Auslesen verbrauchen. ja nein weiß nicht
 10. Versionsverwaltungssysteme werden bei Softwarefirmen nur sporadisch eingesetzt, da sie die Arbeit im Team behindern. ja nein weiß nicht
 11. Versionsverwaltungssysteme werden bei Softwarefirmen nur sporadisch eingesetzt, da nur Programmquellen verwaltet werden können, aber keine Entwurfsdokumente und Testfälle ja nein weiß nicht
 12. Es ist möglich, die Versionen der Quellen aus einem ausgelieferten Programm in Binärformat zu erkennen. ja nein weiß nicht
-

Aufgabe 5:

Was ist eine *pipe* im UNIX Betriebssystem?

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

Welches sind die Unterschiede in der Verarbeitung der folgenden beiden Kommandos?

- a) `who | grep 'root'`
- b) `who > tmp; grep 'root' tmp`

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

Welche Resultate liefern die Kommandos?

.....
.....

Welches Kommando ist vorzuziehen und warum?

.....
.....

Aufgabe 6:

Nach der folgenden Kommandosequenz

```
rm -f -r *  
echo hallo > tmp1  
ln tmp1 tmp2
```

gibt es zwei Einträge in der momentanen Arbeitsdirectory. Erklären sie die unterschiedlichen Effekte der beiden folgenden Befehlssequenzen für diese directory.

1. `cp tmp2 tmp3; rm tmp2`

.....
.....

2. `mv tmp2 tmp3`

.....
.....
