

---

Aufgaben zur Klausur **Unix** im WS 2000/01 (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.

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 zum Durchsuchen aller HTML-Dateien im Verzeichnisbaum des momentanen Arbeitsverzeichnisses. Es sollen für alle diese Dateien die Zeilen ausgegeben werden, in denen H1- oder H2-tags vorkommen. Groß- und Kleinschreibung ist für HTML-tags nicht signifikant.

.....

.....

.....



**Aufgabe 2:**

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 3:

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 `.`, für nicht druckbare Zeichen gibt es Ersatzsequenzen, z.B. für Zeilenvorschub `\n`, für Tabulator `\t`.
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:

Aus Texten sollen Datumsangaben gefiltert werden. Es soll nach Datumsangaben gesucht werden, die aus einer 2-stelligen Tagesangabe bestehen, diese soll durch ein - von der Monatsangabe getrennt sein. Die Monatsangabe ist entweder eine 2-stellige Zahl oder ein 3-buchstabiges Kürzel für den Monat, z.B. `jan`, `feb`, `mar`, . . . Nach der Monatsangabe folgt wieder durch ein - getrennt eine 2- oder 4-stellige Jahreszahl.

Geben Sie einen regulären Ausdruck für dieses Datumsformat an:

.....  
.....

In C++-Programmen sind Kommentare erlaubt, die durch 2 Schrägstriche eingeleitet werden und bis zum Zeilenende gehen.

Geben Sie einen regulären Ausdruck für diese Kommentarart an:

.....  
.....

In HTML-Texten wird mit Tags gearbeitet um Textteile zu markieren. Ein HTML-Tagname besteht aus einer Folge von Buchstaben und Ziffern, außerdem sind - innerhalb von Namen erlaubt. Da erste Zeichen muß ein Buchstabe sein.

Beispiele

korrekt: `H1 HTTP-EQUIV BIG`

falsch: `ABC- 1ABC X--Y`

Geben Sie einen regulären Ausdruck für diese Tagnamen an:

.....  
.....

**Aufgabe 4:**

Was versteht man unter einem *Remote Procedure Call (RPC)*?

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

*RPCs* sind aufwendiger und rechenintensiver als normale Prozeduraufrufe. Wann ist es trotzdem sinnvoll, notwendig oder schneller *RPCs* zu verwenden.

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

**Aufgabe 5:**

Das Kommando `ls` liefert die folgende Ausgabe

```
-r c.1 c1 c1.1 c1.2 c2.* c2.*? c2.1 c2.11 c2.2 c2.?
```

Wie listet man mit dem `ls` Kommando und genau einem Argument mit "wildcards" die Dateinamen `c1.1 c1.2` ?

.....

Wie listet man mit dem `ls` Kommando und genau einem Argument mit "wildcards" die Dateinamen `c2.1 c2.2` ?

.....

Wie löscht man die Datei `c2.*` ?

.....

Wie löscht man die Datei `c2.*?` ?

.....

Wie löscht man die Datei `-r` (`-r` ist eine Option des Kommandos `rm`) ?

.....

Welche der folgenden Programme werten wildcards aus: `bash`, `cat`, `ls`, `more`, `find`?

.....