

Name	Matrikelnummer

# Klausur zur Veranstaltung Software-Entwicklung 2

Sommersemester 2004  
Hessische VWA  
Dr. Alexandra Roder, Bernd Ulmann

Hinweise:

- Die Klausur besteht aus 7 Aufgaben mit insgesamt 15 Teilaufgaben.
- Insgesamt sind 120 Punkte erreichbar, jede Teilaufgabe entspricht 8 Punkten.
- Als Hilfsmittel sind alle „unbelebten Hilfsmittel“, d.h. insbesondere Ihre Vorlesungsmitschriften sowie das Skript, zugelassen.
- Lesen Sie jede Aufgabe vor Beginn ihrer Bearbeitung gründlich (!) durch!
- Bitte schreiben Sie leserlich! (Dankeschön! :-)
- Verwenden Sie bitte keinen (!) Bleistift!
- Vergessen Sie nicht, Primärschlüsselbezeichnungen zu unterstreichen!

## Viel Erfolg!

## Aufgabe 1

In den folgenden Teilaufgaben werden Sie ein kleines Perl-Programm entwickeln, das Ihnen eine Liste natürlicher Zahlen, ihrer Quadrate und Kuben in Form einer einfachen Tabelle ausgibt. Erwartet wird eine Ausgabe, die etwa folgende Form (beispielhaft für die ersten sechs Werte) besitzt:

Zahl	Quadrat	Kubus
1	1	1
2	4	8
3	9	27
4	16	64
5	25	125
6	36	216

1. Schreiben Sie zunächst ein einfaches Perl-Programm mit einer `for`-Schleife, das keine Werte vom Benutzer einliest und bei jedem Aufruf eine Liste (siehe oben) der ersten sechs natürlichen Zahlen mit ihren Quadraten und Kuben generiert.

2. Schreiben Sie das Programm nun so um, daß es vom Benutzer einen Wert einliest, der angibt, bis zu welcher natürlichen Zahl die Tabelle erzeugt werden soll. Das Einlesen des Wertes soll von einem erläuternden Text begleitet werden (Prompt). Sie müssen sich nicht um Sicherheitsabfragen kümmern – gehen Sie der Einfachheit halber davon aus, daß nur sinnvolle Werte eingegeben werden.

3. Schreiben Sie das Programm nun so um, daß es anstelle der zentralen `for`-Schleife eine `while`-Schleife verwendet.

## Aufgabe 2:

Im weiteren sollen Sie einige Fragen zu dem folgenden kleinen Perl-Programm, das absichtlich kommentarlos ist, beantworten:

```
use strict;
use warnings;

my @zeilen;

# Belege das Feld:
$zeilen[0] = 'Zebra';
$zeilen[1] = 'Affe';
$zeilen[2] = 'Loewe';
$zeilen[3] = 'Baer';
$zeilen[4] = 'Pinguin';

my $anzahl = 5;

my $aktion = 1;

while ($aktion)
{
    $aktion = 0;
    for (my $i = 0; $i < $anzahl - 1; $i++)
    {
        if ($zeilen[$i] ge $zeilen[$i + 1])
        {
            ($zeilen[$i], $zeilen[$i + 1]) = ($zeilen[$i + 1], $zeilen[$i]);
            $aktion = 1;
        }
    }
}

for (my $i = 0; $i < $anzahl; $i++)
{
    print "$zeilen[$i]\n";
}
```

1. Welche Ausgabe erzeugt das oben stehende Programm, wenn es mit Hilfe des Perl-Interpreters laufen gelassen wird?

2. Was tut das Programm generell, d.h. was täte es, wenn das Feld `@zeilen` mit anderen Werten gefüllt wäre?

3. Welchen Berechnungsaufwand zieht dieses Programm, bezogen auf die Anzahl von Elementen in `@zeilen` nach sich? D.h. wie verlängert sich seine Laufzeit, wenn `@zeilen` beispielsweise doppelt, dreimal, viermal, etc. so viele Einträge enthält?

### Aufgabe 3:

Lesen (und verstehen) Sie das folgende kleine Perl-Programm:

```
#
# Das folgende Programm soll die Summe aller
# natuerlichen Zahlen von 1 bis einschliesslich
# 100 berechnen und ausgeben.
#

use strict;
use warnings;

my $ende = 100;

for (my $i = 1; $i < $ende; $i++)
{
    $summe = $summe + 1;
}

print "$summe\n";
```

1. In dem obenstehenden Programm sind drei Fehler versteckt – finden und benennen Sie sie (diese Aufgabe zählt entsprechend auch dreifach, d.h. pro gefundenem Fehler erhalten Sie 8 Punkte).

## **Aufgabe 4:**

Beschreiben Sie kurz (in Stichworten) den Unterschied im Focus in der Definitionsbeziehungweise der Entwurfsphase.

## **Aufgabe 5:**

1. Welches Vorgehensmodell würden Sie warum bei der Entwicklung eines Steuerprogrammes für eine Herz-Lungen-Maschine vorschlagen? Bedenken Sie, daß es sich hierbei um eine Applikation handelt, von der Menschenleben abhängen, wobei Fehler nicht tolerierbar sind.

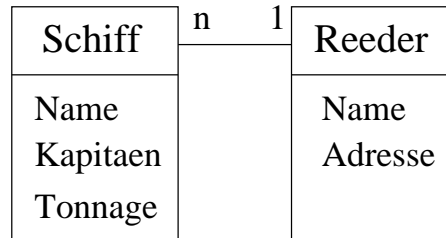
2. Warum würden Sie kein ausschließliches Prototypingmodell in Betracht ziehen (außer, Sie sind in der Bestattungsbranche tätig)?

## **Aufgabe 6:**

1. Erläutern Sie den prinzipiellen Unterschied zwischen SQL und OQL – vor allem hinsichtlich der Mächtigkeit der zur Verfügung stehenden Operationen. Wie muß jener bei der Implementierung eines auf Datenbanken zugreifenden Programmes berücksichtigt werden?

## Aufgabe 7:

1. Ermitteln Sie die notwendigen relationalen Tabellenstrukturen für die Abbildung des folgenden Klassendiagramms mit Hilfe einer relationalen Datenbank:



2. Ermitteln Sie nun die notwendigen relationalen Tabellenstrukturen für das folgende (leicht modifizierte) Klassendiagramm:

