

Holger Füßler

A5, 6, Raum B 219

68131 Mannheim

Telefon: (0621) 181-2605

Email: fuessler@informatik.uni-mannheim.de

Robert Schiele

B6, 29, Raum C0.04

68131 Mannheim

Telefon: (0621) 181-2214

Email: rschiele@uni-mannheim.de

---

Praktische Informatik I  
Wintersemester 2005/2006

1. Scheinklausur  
22. Dezember 2005

---

Name: \_\_\_\_\_

Matrikelnummer: \_\_\_\_\_

Studienfach: \_\_\_\_\_

Semester: \_\_\_\_\_

Tutor: \_\_\_\_\_

---

Hinweise

---

1. Überprüfen Sie bitte Ihr Klausurexemplar auf Vollständigkeit (20 Seiten).
2. Bearbeiten Sie die Aufgaben *ausschließlich* auf dem Aufgabenblatt der jeweiligen Aufgabe.
3. Schreiben Sie auf jedes Blatt, das bewertet werden soll, oben ihren Namen und ihre Matrikelnummer.
4. Verwenden Sie nur dokumentenechte Stifte (z. B. keinen Bleistift) und keine roten Stifte.
5. Es sind keine Hilfsmittel zugelassen.
6. Bearbeitungszeit: 90 Minuten.

---

Korrekturzeile

Bitte *nicht* ausfüllen!

---

1	2	3	4	5	6	7	8	gesamt
4	6	6	6	5	6	6	6	45

Name:

Matrikelnummer:

---

Aufgabe 14 Punkte

---

Die folgenden Klassen Alpha und Beta enthalten Fehler. Listen sie diese mit KlassenName:Zeilennummer auf. Erklären Sie kurz, was schief geht und machen Sie jeweils einen Vorschlag zur Fehlerbehebung. *Hinweis: Achten Sie u.a. auf Zugriffsrechte (Sichtbarkeit) und den Geltungsbereich von Variablen und Methoden. Achten Sie auch auf Typverkleinerungen.*

```
1 public class Alpha {  
    protected static final double fd;  
    private static float f;  
    private double d;  
    static {  
6      fd = 9.0;  
      d = 3.0;  
    }  
    private Alpha(int d) {  
11     this.d = d;  
     Alpha.f = 3.0F;  
    }  
    public abstract String toString();  
}
```

```
1 public class Beta extends Alpha {  
    final int j;  
    public Beta() {  
        super(3.0);  
        i = 9;  
6      j = 9;  
    }  
    private String toString() { return new String("Beta"); }  
}
```

Name:

Matrikelnummer:

---

 Aufgabe 2
 

---

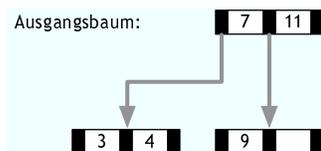
6 Punkte

Im folgenden finden Sie B-Bäume vom Rang  $m = 3$  (jeder Knoten hat höchstens 3 Söhne), auf denen Sie Einfüge- und Löschooperationen durchführen sollen. Schreiben Sie den Lösungsbaum neben den Ausgangsbaum.

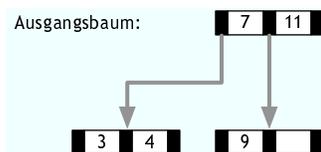
## Aufgabe 2 a)

2 Punkte

Fügen Sie das Element 8 ein.



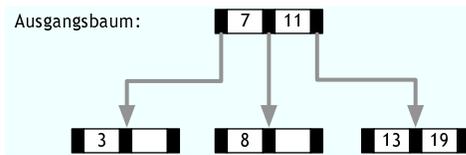
Fügen Sie das Element 5 ein.



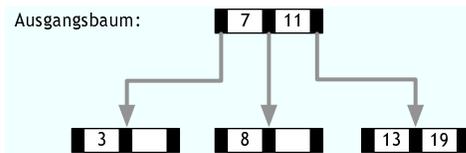
## Aufgabe 2 b)

2 Punkte

Löschen Sie das Element 13.



Löschen Sie das Element 7.



## Aufgabe 2 c)

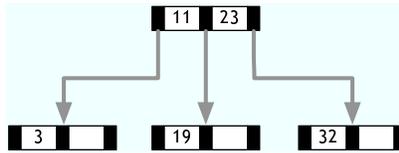
1 Punkte

Was muss passieren, damit sich die Höhe eines B-Baumes erhöht bzw. erniedrigt?  
Begründen Sie kurz.

## Aufgabe 2 d)

1 Punkte

Geben Sie für den folgenden B-Baum eine möglichst kurze Folge von Einfüge- bzw. Löschooperationen an, die die Höhe des Baumes wachsen bzw. schrumpfen lassen.



Name:

Matrikelnummer:

---

Aufgabe 36 Punkte

---

Für die Übungsblätter haben Sie ausführlich die Programme `java` und `javac` kennen gelernt. Wenn man das Programm `java` mit dem Schalter `-h` aufruft, antwortet es mit einer kurzen Hilfe, die dem Benutzer verfügbare Optionen auflistet. Im Folgenden finden Sie einen Auszug dieser Hilfe.

```
Usage: java [-options] class [args...]  
         (to execute a class)  
or java [-options] -jar jarfile [args...]  
         (to execute a jar file)
```

where options include:

```
-classpath <class search path of directories and zip/jar files>  
         A : separated list of directories, JAR archives,  
         and ZIP archives to search for class files.  
-D<name>=<value>  
         set a system property  
-version      print product version and exit
```

Zeichnen Sie aus dieser Information ein Syntaxdiagramm für einen korrekten Kommandozeilenaufruf. Beachten Sie, dass eine beliebige Anzahl von Kommandozeilenargumenten übergeben werden darf. Geben Sie auch notwendige Leerzeichen im Syntaxdiagramm an.

Nehmen Sie folgende Vereinfachungen an:

- Groß- und Kleinschreibung spielt keine Rolle.
- Für ein einzelnes Argument (`[args...]`) dürfen Sie das Nonterminalsymbol `arg` benutzen, das Sie nicht näher definieren müssen.
- Klassennamen bestehen nur aus den Buchstaben `a-z`.
- Jar-files enden auf `.jar` und bestehen ansonsten auch nur aus `a-z`.

Name:

Matrikelnummer:

---

**Aufgabe 4**

---

**6 Punkte****Aufgabe 4 a)****2 Punkte**

Zeichnen Sie schematisch die Datenstruktur einer doppelt verketteten Liste. Aus Ihrer Zeichnung soll hervorgehen, welche Informationen ein Listenelement neben dem eigentlichen Datum noch enthält. Geben Sie den einzelnen Bestandteilen eines Listenelements sinnvolle Namen.

## Aufgabe 4 b)

4 Punkte

Betrachten Sie nun einen Algorithmus, der als Eingabe eine Referenz auf ein Listenelement übergeben bekommt und dieses aus der Liste entfernen soll. Zeichnen Sie für diesen Algorithmus ein Struktogramm. Die Anweisungen im Struktogramm müssen nicht denen einer realen Programmiersprache entsprechen, sondern dürfen in Pseudocode formuliert sein, das heißt, Sie können die Anweisungen in der Art „Setze Referenz A auf B.C“ schreiben.

---

Aufgabe 55 Punkte

---

Geben Sie an, welche Ausgabe das im folgenden angegebene Programm erzeugt, wenn es mit `java Oscar` aufgerufen wird.

Geben Sie für jede Zeile der Ausgabe auch an, weshalb diese ausgegeben wird, indem Sie angeben, welche Methoden jeweils nacheinander aufgerufen werden.

*Hinweis: Zur Bearbeitung könnte es sinnvoll sein, die Vererbungsbeziehungen zu skizzieren.*

```
1 interface Foxtrot {
    String echo = "Uniform";
    void alfa ();
}

6 interface Golf {
    String echo = "Victor";
    void alfa(String juliett );
}

11 class Hotel implements Golf {
    public void alfa(String kilo) {
        System.out.println("Quebec");
    }
    public static void bravo() {
16     System.out.println("Romeo");
    }
    public void charlie() {
        bravo();
    }
21     public static void delta() {
        System.out.println(echo);
    }
}

26 class India extends Hotel implements Foxtrot {
    public void alfa() {
        System.out.println("Sierra");
    }
    public static void bravo() {
31     System.out.println("Tango");
    }
    public void charlie() {
        bravo();
    }
36 }
```

*// bitte wenden...*

```
41 class Oscar {  
    public static void main(String papa[]) {  
        Golf lima = new Zulu(); // HMF: has to be Golf lima = new India();  
        Hotel mike = (India)lima;  
        lima.alfa("X-Ray");  
46     mike.alfa("Yankee");  
        mike.bravo();  
        mike.charlie ();  
        India.delta ();  
    }  
51 }
```

---

**Aufgabe 6**

---

**6 Punkte**

Implementieren Sie eine Methode

`static int[] [] matrixAdd(int m1[] [], int m2[] [])` derart, dass sie die beiden Eingabematrizen `m1` und `m2` addiert und die Ergebnismatrix zurückgibt.

Eine Matrix wird mit einer anderen Matrix addiert, indem die Matrixelemente einzeln addiert werden.

Das heißt: Seien  $A, B$  Matrizen mit  $I$  Zeilen und  $J$  Spalten und  $\alpha_{ij}, \beta_{ij}$  das jeweilige Element in Zeile  $i$  und Spalte  $j$ . Dann ist  $\Gamma = A + B$  gegeben durch  $\gamma_{ij} = \alpha_{ij} + \beta_{ij}$  für alle  $i, j$ . Eine Matrixaddition ist nur möglich, wenn beide Operanden die selbe Anzahl von Zeilen und Spalten besitzen. Stellen Sie sicher, dass Ihre Implementation bei einer ungültigen Eingabe (zum Beispiel ungültige Matrix oder nicht addierbare Matrizen) statt der Ergebnismatrix eine Nullreferenz zurückgibt.

Name:

Matrikelnummer:

---

**Aufgabe 7**

---

**6 Punkte**

Unter der Primfaktorzerlegung einer Ganzzahl  $\nu > 0$  versteht man die Aufspaltung dieser Zahl in  $p$  von 1 verschiedene Primfaktoren  $\pi_1, \dots, \pi_p$ , so dass gilt

$$\nu = \pi_1 \cdot \pi_2 \cdot \dots \cdot \pi_{p-1} \cdot \pi_p$$

Zum Beispiel wäre die Primfaktorzerlegung der Zahl  $210 = 2 \cdot 3 \cdot 5 \cdot 7$ , wohingegen 23 die “Zerlegung” 23 hat, da es sich bereits um eine Primzahl handelt. Zur Erinnerung: eine Primzahl ist eine Zahl, die ganzzahlig außer durch sich selbst nur durch 1 teilbar ist.

**Aufgabe 7 a)****4 Punkte**

Schreiben Sie eine **rekursive** Java-Methode, die die Primfaktoren einer als Argument übergebenen positiven Ganzzahl in aufsteigender Größe ausgibt.

Hinweise:

- Nutzen Sie den Zusammenhang aus, dass nach dem Teilen einer Zahl durch einen ihrer Primfaktoren wieder eine Zahl entsteht, auf die man eine Primfaktorzerlegung anwenden kann.
- Bestimmen Sie einzelne Primfaktoren durch geschicktes “Ausprobieren”.

## Aufgabe 7 b)

1 Punkte

Bei welchen Zahlen  $r$  bekommen Sie die größte Rekursionstiefe im Vergleich mit ähnlich großen Zahlen und wie groß ist diese in Abhängigkeit von  $r$ ? Begründen Sie ohne Beweis.

## Aufgabe 7 c)

1 Punkte

Bei welchen Zahlen  $v$  (wieder im Vergleich mit ähnlich großen Zahlen) sind die meisten Faktor-Tests zu erwarten und wie viele sind es? Begründen Sie ohne Beweis.

---

Aufgabe 86 Punkte

---

Betrachten Sie folgende Methode:

```
public static void sort(double a[], int n) {  
    if (n > 1) {  
        sort(a, n - 1);  
4        while ((--n > 0) && (a[n] < a[n - 1])) {  
            double t = a[n];  
            a[n] = a[n - 1];  
            a[n - 1] = t;  
        }  
9    }  
}
```

## Aufgabe 8 a)

1 Punkte

Beschreiben Sie informal kurz die Funktionsweise des Algorithmusses.

## Aufgabe 8 b)

5 Punkte

Zeigen Sie mit Hilfe einer vollständigen Induktion, dass diese Methode das Feld `a` korrekt aufsteigend sortiert, wenn die Methode mit `sort(a, a.length)` aufgerufen wird.