



Klausur Einführung in die Informatik I für Elektrotechniker

25. Juli 2005

Name:

Matr.-Nr.

Bearbeitungszeit: 120 Minuten

Bewertung (bitte offenlassen :-)

Aufgabe	Punkte	Erreichte Punkte
1	5	
2	7	
3	4	
4	6	
5	4	
6	7	
7	9	
Summe	42	

Spielregeln (**Jetzt lesen!**):

- Benutzen Sie für die Lösung der Aufgaben **nur** das mit diesem Deckblatt ausgeteilte Papier. Lösungen, die auf anderem Papier geschrieben werden, können **nicht** bewertet werden. Schreiben Sie ihre Lösung auch auf die Rückseiten der Blätter; benötigen Sie für eine Lösung mehr als ein Blatt, finden Sie am Ende der Klausur Leerblätter. Zusätzliches Papier können Sie von den Tutoren bekommen.
- Tragen Sie jetzt (vor Beginn der eigentlichen Bearbeitungszeit !!!) auf **allen** Blättern ihren Namen und ihre Matrikelnummer ein.
- Schreiben Sie deutlich! Unleserliche oder zweideutige Lösungen können nicht gewertet werden.
- Schreiben Sie **nicht** mit Bleistift und **nicht** mit rotem oder grünem Stift (das sind die Farben für die Korrektur).
- Lesen Sie die Aufgaben jeweils bis zum Ende durch; oft gibt es hilfreiche Hinweise!
- Kommentare kosten Zeit; kommentieren Sie ihr Programm nur da, wo der Code alleine nicht verständlich wäre.
- Wir weisen noch einmal darauf hin, dass die Benutzung von Taschenrechnern und anderen elektronischen Hilfsmitteln nicht gestattet ist.

Viel Erfolg!

Aufgabe 2 (7 Punkte) Java.

1. (1 Punkt) Aus wievielen Klassen muss ein lauffähiges Java-Programm mindestens bestehen?

2. (2 Punkte) Schreiben Sie eine Methode

```
int f(int x)
```

welche die Berechnung durchführt, die durch folgende Tabelle definiert wird:

x	$f(x)$
1	1
2	4
3	9
4	15

(Hinweis: Das Ergebnis der Funktion für nicht angegebene Werte von x ist nicht festgelegt.)

3. (2 Punkte) Beschreiben Sie in natürlicher Sprache, welche Ausgabe von folgender Java-Methode auf dem Bildschirm erzeugt wird.

```
void f(int n) {
    int x = 1;
    int y = 2;
    int i = n;
    while (i >= 0) {
        Terminal.println(i * n);
        n--;
        x = x + y;
        y--;
        i--;
    }
}
```

4. (2 Punkte) Welche Fehler enthält folgende Java-Klasse? Geben Sie jeweils die Zeilennummer an und beschreiben Sie den Fehler. Folgefehler (also Fehler, die aus anderen Fehlern resultieren) sollen ignoriert werden.

```
1  class Programm {
2      int fib(int x) {
3          return fib(x - 1) + fib(x - 2);
4      }
5      int fab(double x) {
6          return fib(x);
7      }
8      int fub(int[10] x) {
9          return x[0];
10     }
11 }
```

Aufgabe 3 (4 Punkte) Zahlensysteme.

1. (2 Punkte) Rechnen Sie die Zahl $432_{(5)}$ in eine Zahl zur Basis 16 um. Lassen Sie den Lösungsweg erkennen.

2. (2 Punkte) Berechnen Sie die folgenden Aufgaben unter Verwendung der Zweierkomplementdarstellung mit 4 Bit. Lassen Sie den Lösungsweg erkennen. Bei welchen Aufgaben findet ein Über- bzw. Unterlauf statt? Woran erkennen Sie jeweils den Über- bzw. Unterlauf?

(a) $-3 + 3$

(b) $2 + (-6)$

(c) $-2 + (-7)$

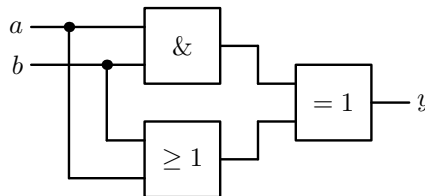
Aufgabe 4 (6 Punkte) Schaltungen.

- (3 Punkte) Entwickeln und zeichnen Sie eine Schaltung, die als An-/Ausschalter für ein elektronisches Spielzeug dienen kann.

Der Eingang e dieser Schaltung hat den Wert 1, wenn der Schalter gedrückt ist und den Wert 0, wenn er nicht gedrückt ist. Bei jedem Drücken des Schalters soll der Ausgang y von 0 auf 1 bzw. von 1 auf 0 umgeschaltet werden, je nachdem, welcher Wert vorher am Ausgang anlag.

(Hinweis: Sie können davon ausgehen, dass der Schalter immer nur ganz kurz gedrückt wird.)

- (2 Punkte) Stellen Sie die Wertetabelle für folgende Schaltung auf:



- (1 Punkt) Zeichnen Sie eine einfachere Schaltung mit der gleichen Funktion wie in der vorigen Unteraufgabe.

Aufgabe 5 (4 Punkte) Numerische Berechnung.

1. (2 Punkte) Schreiben Sie eine Methode

```
double reihensum(int n)
```

welche das n -te Glied der Reihe (s_n) berechnet.

$$s_n = \sum_{i=2}^{n+1} (-1)^i \cdot \frac{1}{i-1}$$

Beispiel:

$$\text{reihensum}(5) = \frac{1}{1} - \frac{1}{2} + \frac{1}{3} - \frac{1}{4} + \frac{1}{5} = 0.783333$$

2. (2 Punkte) Schreiben Sie eine Methode

```
double[] tabelle(int n)
```

welche ein Array der Länge n zurückliefert, wobei die Elemente des Arrays das 1. bis n -te Glied der oben definierten Reihe (s_n) enthalten.

Beispiel:

```
tabelle(5) = {1.0, 0.5, 0.833333333, 0.583333333, 0.783333333}
```

Aufgabe 6 (7 Punkte) Array- und Matrix-Operationen.

1. (3 Punkte) Schreiben Sie eine Methode

```
double[] matrixAnalyse(double[] [] a)
```

welche eine beliebig große rechteckige Matrix als Eingabe erhält und ein Array mit drei Elementen zurückliefert. Das erste Element soll der kleinste Wert der Matrix sein, das zweite der Mittelwert aller Elemente der Matrix, und das dritte der größte Wert. Sie können davon ausgehen, dass die Eingabematrix mindestens eine Zeile und eine Spalte hat.

Beispiel: Für eine Matrix $x = \begin{pmatrix} 1 & 2 \\ 3 & 0 \end{pmatrix}$ ergibt ein Aufruf der Methode `matrixAnalyse(x)` das Array `{0, 1.5, 3}`.

2. (2 Punkte) Schreiben Sie eine Methode

```
int runs(float[] b)
```

welche ein beliebig langes Array von `float`-Zahlen als Eingabe erhält und die Anzahl von *Runs* in dem Array `b` zählt. Ein Run ist eine Folge von aufsteigenden oder gleichbleibenden Werten, d.h., jeder Übergang von einem größeren zu einem kleineren Wert beginnt einen neuen Run. Sie können davon ausgehen, dass das Eingabearray mindestens ein Element enthält.

Beispiel:

```
runs({1.1f, 2.1f, 4.2f, 1.0f, 2.5f, 0.0f}) ergibt 3  
runs({0.0f, 1.1f}) ergibt 1
```

3. (2 Punkte) Schreiben Sie eine Methode

```
int vorkommnisse(String[][] b, String s)
```

welche berechnet, wie oft der String `s` als Element in der Matrix `b` vorkommt. Sie können davon ausgehen, dass die Matrix nicht leer ist.

Aufgabe 7 (9 Punkte) Kassensystem.

In dieser Aufgabe soll ein Teil eines Kassensystems programmiert werden, wie es zum Beispiel in einem Supermarkt eingesetzt werden kann.

Das Programm erhält als Eingabe die Binärdarstellung einer Produktnummer, die von einem Strichcodeleser stammen könnte. Diese Binärdarstellung muss in eine ganze Zahl umgewandelt werden. Danach muss nach dieser Nummer in einer Liste von Waren gesucht werden und schließlich der Name des eingescannten Produktes ausgegeben werden.

Die folgenden Aufgaben sollen die einzelnen Schritte realisieren.

Für alle folgenden Teilaufgaben gilt die Definition der folgenden Klasse, welche die verwalteten Waren des Systems darstellt:

```
class Ware {
    int nummer;
    String name;
    Ware(int nummer, String name) {
        this.nummer = nummer;
        this.name = name;
    }
}
```

1. (3 Punkte) Schreiben Sie eine Methode

```
int strichCode(boolean[] code)
```

welche einen Produktcode (dargestellt als Array von Boolean-Werten) in eine Ganzzahl umwandelt. Ein Eintrag **true** steht dabei für eine 1, ein Eintrag **false** für eine 0. Das Array enthält die höherwertigen Stellen des Codes am Anfang, die niederwertigen Stellen am Ende. Sie können davon ausgehen, dass das Array `code` nicht leer ist.

Beispiel:

```
strichCode({true, false, false, false, true, false}) ergibt 34
strichCode({true, false, true, true}) ergibt 11
```

2. (2 Punkte) Gehen Sie davon aus, dass das Kassensystem die verwalteten Waren in einem Attribut speichert, das folgendermaßen deklariert ist:

```
Ware[] waren;
```

Schreiben Sie eine Methode

```
Ware suche(int num)
```

welche in dem Attribut `waren` nach dem Produkt mit der Nummer `num` sucht und die Ware als Ergebnis zurückliefert. Sie können davon ausgehen, dass nur gültige Produktnummern eingegeben werden.

3. (2 Punkte) Schreiben Sie eine Methode

```
void hinzufuegen(int nummer, String name)
```

welche dem Warensortiment eine Ware mit der gegebenen Produktnummer und dem gegebenen Namen hinzufügt. Dafür soll das Array, das in dem Attribut `waren` gespeichert ist (siehe vorige Unteraufgabe), um einen Eintrag erweitert werden

4. (2 Punkte) Schreiben Sie nun eine Methode

```
void kassieren(boolean[] code)
```

welche einen Warencode als Eingabe bekommt, nach dem dazugehörigen Wareneintrag sucht und den Namen des Produkts auf dem Bildschirm ausgibt. Sie können davon ausgehen, dass nur gültige Codes eingegeben werden.



Fak. ET/Inform.
Klausur InfET I
25. Juli 2005

Name:

Matr.-Nr.



Fak. ET/Inform.
Klausur InfET I
25. Juli 2005

Name:

Matr.-Nr.
