

Modellierung und Programmierung 1
Übungsserie 1

Abgabetermin: 02.11.2014, 23:55 Uhr

Grundsätzlich sind Nebenrechnungen anzugeben und Antworten zu begründen.
Einzureichen sind, bei mehreren Dateien als .zip-Archiv:
Lösungen als .pdf-Datei, Programme als Quellcode.

1. Konvertierung

Vervollständigen Sie (ohne Angabe der Nebenrechnung) die nachfolgenden Tabellen:

a) Natürliche Zahlen

Basis 2	Basis 10	Basis 16
	50 242	
11 0100 0100		
		CA FE

b) Rationale Zahlen

Basis 2	Basis 10	Basis 16
	13.812 5	
101.011		
		1B.68

c) Bestimmen Sie b bzw. x so, dass nachfolgende Gleichungen erfüllt sind:

$$(123)_b = (198)_{10}$$

$$(0.\overline{22})_3 = (x)_{10}$$

2. Maschinenzahlen

a) Berechnen Sie die Codierung von -123 als `int`-Maschinenzahl und die Codierung von 123.515625 als `float`-Maschinenzahl.

(`float` $\hat{=}$ `single` nach IEEE-Standard)

b) Gegeben sei die folgende Codierung:

1000 0000	0000 0000	0000 0000	0000 0010
-----------	-----------	-----------	-----------

Welche `int`- bzw. welche `float`-Maschinenzahl wird durch diese Bitfolge dargestellt?

c) Geben Sie, ohne Berücksichtigung der Sonderfälle mit denormalisierter Mantisse, die größte und die kleinste darstellbare positive `float`-Maschinenzahl als Bitfolge und als Dezimalzahl an. Berechnen Sie den für diesen Zahlbereich größten und kleinsten möglichen Rundungsfehler.

Hinweis: Führen Sie die Fehlerberechnungen von den Bitfolgen ausgehend, zunächst mit Zweierpotenzen durch.

3. Algorithmen und Programme

- a) Beweisen oder widerlegen Sie:
Für jede natürliche Zahl n ist $f(n) = n^2 + n + 41$ eine Primzahl.
- b) Entwerfen Sie mit der in der Vorlesung vorgestellten minimalen Programmiersprache ein Programm $prim(n)$ mit

$$prim(n) = \begin{cases} 1, & \text{falls } n \text{ eine Primzahl ist,} \\ 0, & \text{sonst.} \end{cases}$$

Achten Sie auf die Effizienz Ihres Programmes. Protokollieren Sie die Belegungsänderungen aller im Programm vorkommenden Variablen für $prim(37)$ und $prim(679)$ tabellarisch.

4. JavaScript

Verändern Sie a) das Hallo-Welt-Applet und b) die dazugehörige HTML-Seite aus der Vorlesung so, dass man mittels Mausklick die Vorder- und Hintergrundfarben vertauschen kann.

Hinweise:

Bitte beachten Sie, dass nicht alle Browser JavaScript unterstützen. Verwenden Sie zum Beispiel Mozilla Firefox 24 oder Internet Explorer 11.

- Als Farben können Sie die Klassenkonstanten der Klasse **Color** verwenden, zum Beispiel: **Color.white** für Weiß und **Color.black** für Schwarz.
- Speichern Sie die aktuelle Vordergrundfarbe und die aktuelle Hintergrundfarbe in dafür vorgesehene Farbvariablen, z.B. **Color vordergrundFarbe = Color.white;**
- Mit **setForeground(Color);** und **setBackground(Color);** können die aktuellen Werte innerhalb der Hauptmethode **paint** gesetzt werden.
- Um die Farben zu wechseln, vertauschen Sie in einer speziellen Methode **set** die Werte der Farbvariablen.
- **onClick** ist die Java Script Anweisung im HTML-Code, mit der man einen Mausklick abfragen kann. Jeder Mausklick ruft dann die Methode **set** zum Farbenwechseln auf.

HalloWeltAppletPlus.java (Grobstruktur)

```
import java.applet.*;                                     //Applet
import java.awt.*;                                       // Graphics, Color
public class HalloWeltAppletPlus extends Applet
{
    // Vordergrundfarbe und Hintergrundfarbe vereinbaren

    public void paint( Graphics g)
    {
        // Vordergrundfarbe und Hintergrundfarbe setzen
        // Bildschirmausschrift erzeugen

    }
    public void set()
    {
        // Farben wechseln

        repaint( 100L);
    }
}
```