

## Übung 04: Gleitkommazahlen und Methoden

Abgabetermin: 10. 11. 2009

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

Gruppe: G1 (Prähofer)  G2 (Wolfinger)  G3 (Wolfinger)  G4 (Jahn)

Aufgabe	Pkte	Abzugeben	gelöst	Erreichte Pkte	Kommentar
Aufgabe 04.1	12	Prosabeschreibung Ablaufdiagramm Java-Programm Testfälle und Ergebnisse	<input type="checkbox"/>		
Aufgabe 04.2	12	Prosabeschreibung Ablaufdiagramm Java-Programm Testfälle und Ergebnisse	<input type="checkbox"/>		

### Aufgabe 4.1: Näherungsberechnungen zur Bestimmung der Kreiszahl $\pi$

Im Laufe der Zeit entwickelten Mathematiker verschiedene Näherungsverfahren zur Bestimmung der Kreiszahl  $\pi$ . Schreiben Sie ein Java-Programm, das die unten angeführten Näherungsverfahren implementiert.

- Wallissches Produkt (seit 1655):

$$\frac{\pi}{2} = \prod_{n=1}^{\infty} \frac{2n}{2n-1} * \frac{2n}{2n+1} = \frac{2}{1} * \frac{2}{3} * \frac{4}{3} * \frac{4}{5} * \frac{6}{5} * \frac{6}{7} * \frac{8}{7} * \frac{8}{9} \dots$$

- Baily-Borwein-Plouffe-Formel (seit 1996):

$$\pi = \sum_{n=0}^{\infty} \frac{1}{16^n} \left( \frac{4}{8n+1} - \frac{2}{8n+4} - \frac{1}{8n+5} - \frac{1}{8n+6} \right)$$

Startet der Benutzer das Programm, soll er gefragt werden wie weit der näherungsweise berechnete Wert von der Java-Konstante `Math.PI` abweichen darf. Anschließend berechnet das Programm  $\pi$  mit den zwei Verfahren, bis die gewünschte Genauigkeit `Epsilon` erreicht wird. Eine Funktion mit folgendem booleschen Ausdruck soll als Abbruchbedingung der Berechnungsschleifen dienen:

```
Math.abs(Math.PI - calculatedPi) <= epsilon
```

Deklaren Sie eine globale Konstante `PRINT_OUTPUT` vom Datentyp `boolean`. Der Wert dieser Konstante bestimmt, ob bei den Berechnungen von  $\pi$  die Anzahl der benötigten Schleifendurchläufe und ein Vergleich zwischen der Konstante `Math.PI` und dem errechneten  $\pi$  ausgegeben wird.

Beispiel:

```
C:\GdP\04>java CalculatePi
Bitte Fehlerschranke fuer Pi-Berechnung eingeben: 0.000001
```

Wallissches Produkt:

```
-----
Schleifendurchläufe: 785397
Java Pi:             3,1415926536
Calculated Pi:      3,1415916536
```

Baily-Borwein-Plouffe-Formel:

```
-----
Schleifendurchläufe:      4
Java Pi:                 3,1415926536
Calculated Pi:          3,1415924576
```

```
C:\GdP\04>
```

Hinweise:

- Implementieren Sie die Näherungsverfahren in Funktionen und die Ausgabe in einer Prozedur.
- Die Potenzen in den Reihen sollen "nebenbei" aus dem jeweiligen Vorgängerwert berechnet werden.
- Das Ergebnis einer Ganzzahl-Division ist vom Typ `int`, dabei werden Nachkommastellen abgeschnitten. Um das zu vermeiden muss man vor der Division einen der beiden Operanden in eine Gleitkommazahl verwandeln.

```
double result1 = 3 / 2;           // result1 = 1
double result2 = (double) 3 / 2; // result2 = 1.5
```

Abzugeben ist:

- Die Beschreibung des Programms in Prosa
- ~~▪ Das Ablaufdiagramm~~ **aktualisiert !**
- Das Java-Programm
- Testfälle: eingelesene Werte und Ergebnisse

## Aufgabe 4.2: Statistik über die Anzahl der Mensagäste

Die Mensa einer Universität hat an verschiedenen Wochentagen die Anzahl der Gäste gezählt und die Ergebnisse in eine Datei gespeichert. Für jeden Wochentag ist eine Messfolge wie folgt angegeben:

- Der erste Wert ist der Name des Wochentags
- Es folgen die jeweilig gezählten Mensagäste mit dem Zeichen # als Prefix
- Am Ende jeder Messfolge steht ein Semikolon

Folgend eine Beispieldatei mit den Messwerte:

```
Monday    #4382 #9083 #7892 #5283 #7823 #2397;
Tuesday   #6293 #2309 #6230 #4993 #2892 #3892 #5546;
Wednesday #2389 #2389 #4893 #2892 #3894;
Thursday  #2398 #3023 #3022 #3034 #3200 #3448;
Friday    #2349 #1123 #1218 #1238 #1238 #2122;
Saturday  #984  #581;
Sunday    ;
```

Schreiben Sie ein Java-Programm, das die Messfolgen von einer Datei einliest und für jeden Tag eine Statistik mit folgenden Berechnungen ausgibt:

- Anzahl der Messwerte
- die drei größten Besucherzahlen
- die kleinste Besucherzahl
- die durchschnittliche Besucherzahl  $\bar{x} = \frac{1}{n} * \sum_{i=1}^n x_i$
- die Spannweite der Besucherzahl  $r = x_{max} - x_{min}$

### Hinweise:

Verwenden Sie die Klasse `In`, um die Werte aus der Datei einzulesen. Dazu müssen Sie die Datei mit `In.open()` öffnen und nach dem Einlesen mit `In.close()` schließen.

```
In.open("measurements.txt");
...
In.close();
```

Der Bildschirmdialog des Programms könnte wie folgt aussehen:

```
E:\C_Workspace\GdP\04>java MensaStatistics
```

```
Day:                Monday
-----
Anzahl der Werte:   6
Minimum:            2397
Maximum 1:          9083
Maximum 2:          7892
Maximum 3:          7823
Mittelwert:         6143,33
Spannweite:         6686
...

```

### Abzugeben ist:

- Die Beschreibung des Programms in Prosa
- ~~Das Ablaufdiagramm~~ **aktualisiert!**
- Das Java-Programm
- Testfälle: eingelesene Werte und Ergebnisse