

Prof. Dr. Claudia Müller-Birn, Barry Linnert

Konzepte der imperativen und objektorientierten Programmierung SoSe 2017 Übungsblatt 09

Abgabe: 28.06.2017

Geben Sie immer die Namen aller Mitglieder Ihrer Gruppe an!

Lösungen für Programmieraufgaben sind zu verifizieren oder zu testen.
Beides ist entsprechend zu dokumentieren.

1 Objekte in Java

(18 Punkte)

Schreiben Sie ein Programm in Java, welches Häuser in einer Straße verwaltet. Überlegen Sie sich welche Attribute für ein Haus sinnvoll sind. Setzen Sie auf Basis Ihrer Überlegungen (mindestens) folgende Teilaufgaben um:

- Erstellen Sie eine Klasse "Haus" und eine Klasse "Strasse".
- Die Klasse "Haus" solle alle Attribute zur Verwaltung eines Hauses beeinhalteten. Implementieren Sie zudem Methoden, welche die Anzahl der Wohnungen im Haus, die Anzahl der bewohnten Wohnungen, die Anzahl der unbewohnten Wohnungen zurückgeben.
- Sobald ein Haus ein vorgegebenes Alter erreicht hat und alle Wohnungen unbewohnt sind, soll es abgerissen werden.
- Die Klasse "Strasse" soll eine feste, vorgegebene Anzahl von Häusern verwalten können. Sobald ein Haus abgerissen wurde, soll an dieser Stelle ein neues errichtet werden können. Hierfür sind entsprechende Methoden zu implementieren. Achten Sie darauf, dass kein Haus an einer Stelle eines bestehenden Hauses gebaut werden kann.
- Gibt es noch freie Grundstücke in der Straße, soll an deren Stelle ein Haus gebaut werden, wenn alle Wohnungen in den bestehenden Häusern bewohnt sind.
- Simulieren Sie die Verwaltung der Häuser der Straße. Ein Simulationsschritt beinhaltet die Änderungen an den Häusern mittels einer geeigneten Eingabe für jeweils ein Jahr. Danach schreitet die Zeit für ein Jahr voran.

2 Arithmetik in Java

(12 Punkte)

Die folgende Funktion zeigt acht verschiedene Arten, die Summe $\sum_{i=1}^n \frac{1}{i^2}$ zu berechnen, die für $n \rightarrow \infty$ (langsam) gegen $\frac{\pi^2}{6}$ konvergiert.

```
1 static double Summe(int n) {  
2     double s = 0.0;  
3     float eins = 1;  
  
5     for (int i=1; i<=n; i++)  
6         Variante a bis h  
  
8     return s;  
9 }
```

Varianten:

- a) `s += 1/i*i;`
- b) `s += 1/(i*i);`
- c) `s += 1.0/i/i;`
- d) `s += 1.0/(i*i);`
- e) `s += 1.0/((double)(i)*i);`
- f) `s += 1/(double)(i*i);`
- g) `s += eins/i/i;`
- h) `s += 1/(1.0*i*i);`

Testen Sie jede einzelne dieser Varianten jeweils für $n = 100$, $10\,000$, 10^6 , 10^8 und 10^9 und vergleichen Sie das Ergebnis mit dem Grenzwert. Erklären Sie, was passiert. Formulieren Sie eine Vermutung, wie groß ungefähr der Abstand zwischen dem Grenzwert und der (exakten) Summe der ersten n Glieder ist.