

Präsenzübungen zur Vorlesung
Objektorientierte Programmierung: Wintersemester 2021/2022
Nr. 10

Aufgabe 10.1: Baumbart

Betrachten Sie den folgenden Code-Ausschnitt:

```
1  public abstract class Tree {
2      protected int height;
3
4      public Tree(int height) {
5          this.height = height;
6      }
7
8      public abstract Color getExpectedColor(Date date);
9
10     @Override
11     public String toString() {
12         return "The tree is " + height/100.0 + " meters high.";
13     }
14 }
```

- a) Implementieren sie die Klasse `ChristmasTree`, die die Klasse `Tree` erweitert. Die Klasse soll zusätzlich ein Feld für die Höhe des Sterns auf der Spitze besitzen. Sie können außerdem vereinfacht annehmen, dass unsere Weihnachtsbäume zu den immergrünen Nadelbäumen gehören.
- b) Überschreiben Sie die `toString`-Methode. Diese soll die Höhe des Baums ohne Stern gefolgt von der Höhe mit Stern zurückgeben. Verwenden Sie hier die Methode der Oberklasse!

Aufgabe 10.2: Exceptional Contingency

- a) Implementieren Sie eine Methode `double pyramidVolume(double a, double h)`, die das Volumens einer Pyramide berechnet. Das Volumen einer Pyramide kann durch $\text{Grundflaeche} * \text{Hoehe} / 3$ berechnet werden.
- b) Implementieren Sie als nächstes eine Klasse `NegativeNumberException`, die die Klasse `Exception` erweitert. Sorgen Sie dafür, dass eine aussagekräftige Fehlermeldung auf der Konsole ausgegeben wird.
- c) Erweitern Sie Ihre Methode `pyramidVolume` dahingehend, dass Ihre neue `Exception` geworfen wird, sobald eine negative Zahl übergeben wird.
- d) Handelt es sich bei Ihrer neuen `Exception` um eine geprüfte oder eine ungeprüfte `Exception`? Worin liegt überhaupt der Unterschied?