
Software Engineering

<http://proglang.informatik.uni-freiburg.de/teaching/swt/2009/>

Aufgabenblatt 4

Aufgabe 1: Design by Contract (7 Punkte)

Auf der Homepage zur Vorlesung finden Sie den Code einer Klasse `Stack`, die Stapel implementiert. Leider fehlen alle Vor- und Nachbedingungen und auch die Invarianten. Bitte fügen Sie diese zum Code hinzu.

Aufgabe 2: Design by Contract und Vererbung (9 Punkte)

Betrachten Sie die folgende Schnittstelle `Map`, die Schlüssel (vom Typ `K`) auf Werte (vom Typ `V`) abbildet.

```
interface Map<K,V>
{
    boolean containsKey(K key);
    V get(K key);
    void put(K key, V value);
}
```

Neben den üblichen Anforderungen an die Methoden von `Map` (see <http://java.sun.com/j2se/1.5.0/docs/api/java/util/Map.html>), machen wir die zusätzliche Einschränkung, dass (i) Schlüssel die gleich `null` sind nicht unterstützt werden und (ii) die Methode `get` den Wert `null` zurückliefert falls der angegebenen Schlüssel nicht existiert.

1. Fügen Sie passende Vor- und Nachbedingungen zu den Methoden der Schnittstelle hinzu.
2. Leiten Sie zwei Unterschnittstellen von `Map` ab. Beide Unterschnittstellen sollen die existierenden Vor- und Nachbedingungen der Methoden von `Map` überschreiben. Eine der Unterschnittstellen soll die Methoden korrekt spezialisieren, die Andere soll eine ungültige Methodenspezialisierung enthalten. Bitte begründen Sie Ihre Ergebnisse.