Prof. Dr. Franz Schweiggert
Norbert Heidenbluth
Johannes Mayer

Abteilung Angewandte Informationsverarbeitung

23.6.2004
Blatt 7

Objektorientierte Programmierung mit Java (SS 2004)

1 Wir testen den S.U.R.P.!

Nehmen wir an, unser S.U.R.P. sei wie folgt implementiert:

```
public class Surp {
    // legt einen S.U.R.P. der angegebenen Maximal-Groesse an
    public Surp(int maxSize);
    // liefert die momentane Anzahl der Elemente (<= Maximal-Groesse)
    public int size();
    // ist der S.U.R.P. leer?
    public boolean isEmpty();
    // entfernt und liefert das erste Element
    public Object shift();
    // entfernt und liefert das letzte Element
    public Object pop();
    // haengt o vor den S.U.R.P. => o ist neues erstes Element
    public void unshift(Object o);
    // haengt o hinter den S.U.R.P. => o ist neues letztes Element
    public void push(Object o);
    // liefert das index-te Element des S.U.R.P.
    public Object get(int index);
    // aendert das index-te Element in o
    public void put(int index, Object o);
}
```

Wird versucht ein Surp-Objekt mit Maximalgröße kleiner eins zu erzeugen, so führt dies zu einer IllegalArgumentException. Ist der Surp voll und es wird versucht (mittels push() oder unshift()) ein neues Element hinzuzufügen, so wird eine SurpOverflowException „geworfen". Bei dem Zugriff auf einen nicht vorhandenen Index gibt es eine SurpIndexOutOfRangeException.

Ein Testfall besteht immer aus einer Eingabe, einer erwarteten Ausgabe und einer tatsächlichen Ausgabe, sowie den Zuständen von Objekten, etc. vor und nach der Testausführung. Vor dem Test überlegt man sich die Eingaben, Zustände und das erwartete Verhalten. Nach dem Test überprüft man, ob das erwartete Resultat dem tatsächlichen entspricht.

Implementieren Sie nun mit Hilfe von JUnit, das in den Übungen vorgestellt wird, eine Testklasse SurpTest, die eine Reihe von Testfällen enthält, um u. A. Folgendes zu testen:

- Was passiert bei Angabe einer Maximalgröße von 0 bzw. eines negativen Wertes?
- Was passiert bei leerem S.U.R.P. bzgl. den Operationen size(), isEmpty(), pop(), shift()?
- Was passiert bei vollem S.U.R.P. bzgl. den Operationen push() und unshift()?
- Erhält man mit pop() das, was man mit push() reingesteckt hat? Was ist mit shift() und unshift()?
- Liefert get() tatsächlich das richtige Element? Setzt put() tatsächlich das richtige Element?
- Wird durch pop() bzw. shift() tatsächlich die Referenz aus dem S.U.R.P. entfernt?

Obige Testfälle sollen nicht nur die korrekte Verwendung überprüfen, sondern auch sehen, was bei falscher bzw. nicht erwarteter Eingabe passiert.

Viel Erfolg!