

Programmier-Klausur

Programmieren 2 – PZR 1 (26.1.2021)

Erlaubte Hilfsmittel:

- Rechner (auch der eigene)
- IDE (IntelliJ, Netbeans, Eclipse)
- Dokumentationen Java (auch die Online-Javadoc von Oracle)
- Nur bei der Abgabe: Webbrowser um die Lösungen in Moodle einzustellen

Ablauf:

Dauer: 90 Minuten

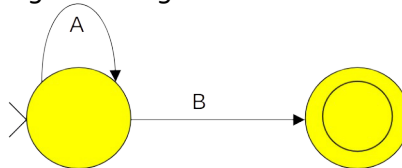
1. Lesen Sie die Aufgabe aufmerksam!
2. Implementieren Sie die Aufgabe!
3. Zippen Sie Ihren Projektordner.
4. Stellen Sie das gezippte File in Moodle ein.

Fragen? Fragen Sie so, dass alle die Frage und die Antwort hören können.

Aufgabe:

Implementieren Sie einen endlichen deterministischen Automaten. Ein solcher Automat startet in einem Startzustand. Dann kann man ihm Zeichen übergeben. Wir nehmen einen String als Datentyp dieser Zeichen. Ein solcher Automat reagiert auf drei mögliche Arten auf ein solches Zeichen: 1) Er verbleibt im gleichen Zustand, 2) er wechselt in einen anderen Zustand oder 3) er dokumentiert einen Fehler, weil das Zeichen in dem aktuellen Zustand nicht erlaubt ist.

Einer oder mehrere Zustände eines solchen Automaten werden als Endzustände definiert. Gelangt der Automat in einen solche Zustand dann gelten alle gelesenen Zeichen als gültiger Satz einer Grammatik.



Das folgende Beispiel zeigt einen endlichen Automaten. Der Startzustand (links) ist mit einem kleinen Haken symbolisiert. Der Endzustand (rechts) ist durch den inneren schwarzen Ring symbolisiert. Zustände, die keinen solchen Haken haben oder keinen inneren Ring wären weitere Zustände des Automaten (nicht in der Abbildung). Dieser endliche Automat würde zum Endzustand gelangen, wenn eine beliebige Anzahl (auch 0) des Symbole A eingegeben werden gefolgt von einem einzelnen B.

Sie sollen einen solchen Automaten implementieren. Ihre Implementierung soll folgende weitere Eigenschaften haben:

1. Sobald bemerkt wird, dass keine Eingabe mehr zu einem gültigen Satz führt, soll der Automat die Eingabe weiterer Zeichen verweigern.
2. Sobald der Automat einen Endzustand angenommen hat, soll er die Eingabe weiterer Zeichen verweigern.
3. Es soll prüfbar sein, ob ein Endzustand erreicht wurde. Ist das der Fall, soll der Automat die Zeichenkette zurück geben können (nicht mit `println` ausgeben) , die als gültiger Satz erkannt wurde.
4. Es soll prüfbar sein, ob der Automat weitere Zeichen annimmt.

Per E-Mail (Ihr HTW-Account) wird Ihnen ein konkreter endlicher Automat zugesendet. Diesen sollen Sie nun implementieren. **Arbeiten Sie test-getrieben!**

