

## 1 Nicht-rekursive Suche (2 Punkte)

Implementiere auf Grundlage der in der Vorlesung vorgestellten Beispielimplementierung nichtdeterministischer Automaten (zweite Version) eine nicht-rekursive Variante der Methode `rcgnz`.

## 2 Epsilon-Abschluss (2 Punkte)

In der Beispielimplementierung nichtdeterministischer Automaten (zweite Version) wird der  $\varepsilon$ -Abschluss zur Laufzeit und u.U. mehrfach berechnet. Implementiere eine Methode `removeEpsilon`, die einen neuen Automaten ohne  $\varepsilon$ -Übergänge berechnet, der die gleiche Sprache wie der ursprüngliche Automat akzeptiert.

## 3 Generieren mit endlichen Automaten (2 Punkte)

Mit endlichen Automaten kann man nicht nur testen, ob eine Zeichenkette erkannt (akzeptiert) wird, man kann Automaten auch verwenden, um alle Wörter einer Sprache aufzuzählen. Implementiere eine Methode `generate()`, die einen Iterator über alle vom Automaten akzeptierten Wörter berechnet:

```
for wort in regex("a(ba|c)*a").nfa().generate():
    print(wort)
```

=> aa, abaa, aca, ababaa, abaca, acbaa, ...

Der Iterator ist in diesem Fall natürlich unendlich. Es bietet sich daher an, den Iterator so zu implementieren, dass der Automat per Breitensuche traversiert wird. Alternativ kann der Iterator auch mit Tiefensuche implementiert werden, wobei in diesem Fall die Suche per Argument so eingeschränkt werden sollte, dass jeder Zustand maximal `n` mal besucht wird.

**Abgabe bis Donnerstag, 2011-05-20, 08:30 Uhr**