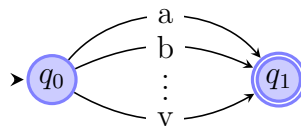


1 Konstruktion eines nichtdeterministischen Automaten aus einem regulären Ausdruck (5 Punkte)

Implementieren Sie den in der Vorlesung vorgestellten Algorithmus zur Konstruktion eines nichtdeterministischen Automaten. Der Parser-Code ist in `RegexParser.java` bereits vorhanden, der Parser wurde mit `bison` aus `RegexParser.y` erzeugt. Sie müssen nur die fünf Methoden in `FiniteAutomaton.java` korrekt implementieren. Sie sollten sich die Dokumentation der Methoden in der Oberklasse `FiniteAutomatonBase.java` ansehen. Die Methode `newCharSetAutomaton` soll einen Automaten für eine Zeichenmenge erzeugen, wie es sie auch in Java-Regex gibt. Das ist ähnlich zu der Alternative, nur dass die ϵ -Kanten nicht nötig sind. Der Automat hat dann folgende Form für z.B. `[a-v]`:



Die Klasse `SubAutomaton` ist als innere Klasse von `FiniteAutomatonBase` realisiert und speichert nur Start- und Endzustand der erzeugten Teilautomaten. Das umgebende `FiniteAutomaton` Objekt übernimmt die Verwaltung des Graphen und des Alphabets. Sie brauchen für die Aufgabe auch die Graph-Klassen aus der vorletzten Übung, die als `jar`-Bibliothek im `template` enthalten ist.

Die `CharFsaTest`-Klasse dient zum Testen des Codes; sie benutze den oben erwähnten Parser zum einlesen von regulären Ausdrücken, und das `graphviz`-Paket um die erzeugten `.dot` Dateien in verschiedene Graphikformate zu layouten.

2 Determinieren eines NFA (6 Punkte)

Implementieren Sie die Determinierung, die auf den Folien beschrieben ist, indem Sie die `determinize`-Methode von `FiniteAutomaton` implementieren. Lesen Sie die Dokumentation in der Oberklasse!

3 Lexikonautomat (4 Punkte)

Laden Sie sich von <http://wortschatz.uni-leipzig.de/html/wliste.html> die 10000 häufigsten Wörter in der Sprache ihrer Wahl herunter und konstruieren Sie einen Automaten aus diesen Wörtern. Schreiben Sie eine Methode, die den Automaten für ein beliebiges eingegebenes Wort traversiert und `true` zurückgibt, wenn das Wort in der Liste ist, sonst `false`.

Testen Sie ihren Code mit der `LexFsaTest`-Klasse.

Abgabetermin ist der 24. Januar

Bitte nur in gewohnter Form an steffen@dfki.de