

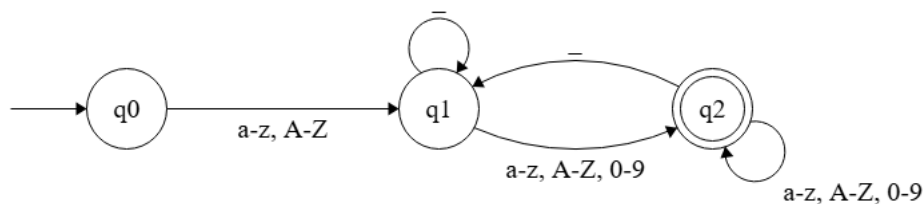
Übungsblatt 01

A1.1: Sprachen von regulären Ausdrücken

- $a + a(a + b)^*a$
 - $L = \{a\} \cup \{a(a + b)^*a\}$
 - $L = \{w \in \{a, b\}^* \mid w \text{ beginnt mit und endet mit } a\}$

A1.2: Bezeichner in Programmiersprachen

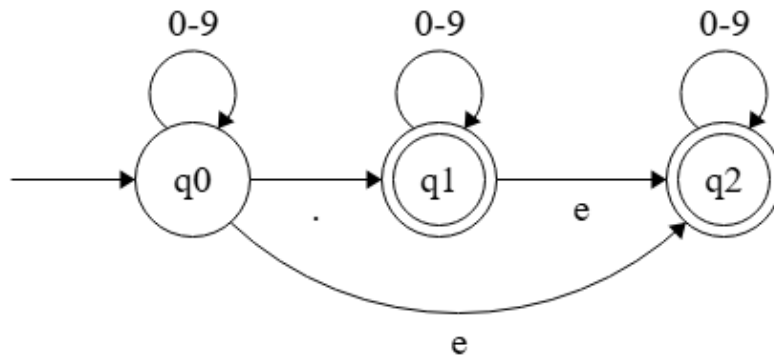
- Regex: $(a-z + A-Z)(a-z + A-Z + 0-9 + _)^*(a-z + A-Z + 0-9)$
 - `NotLocal_1` → "P" erlaubt, danach ist "notLocal_" innerhalb des mittleren partiellen Ausdrucks und am Ende steht "1" was kein Unterstrich ist, das Wort ist außerdem mind. zwei Zeichen lang
 - `vLocal` → "v" erlaubt, danach "Loca" innerhalb des mittleren partiellen Ausdrucks und am Ende steht "l" was kein Unterstrich ist, das Wort ist außerdem mind. zwei Zeichen lang
- DFA
- 5-Tupel $A = \{Q, \Sigma, \delta, q_0, F\}$
 - $Q = \{q_0, q_1, q_2\}$
 - $\Sigma = \{a-z, A-Z, 0-9, _ \}$
 - $F = \{q_2\}$
 - δ :



- vL_o_o_p
 - $q_0 \rightarrow q_1 \rightarrow q_2 \rightarrow q_1 \rightarrow q_2 \rightarrow q_1 \rightarrow q_2 \rightarrow q_1 \rightarrow q_2$
- kAloof_5
 - $q_0 \rightarrow q_1 \rightarrow q_2 \rightarrow q_2 \rightarrow q_2 \rightarrow q_2 \rightarrow q_2 \rightarrow q_1 \rightarrow q_2$
- Grammatik
 - Für eine bessere Übersicht habe ich griechische Symbole als Nichtterminale gewählt
 - 4-Tupel $G_r = \{N, T, P, \Gamma\}$
 - $N = \{\Gamma, \Delta, \Lambda\}$
 - $T = \{a-z, A-Z, 0-9, _ \}$
 - $P :$
 - $\Gamma \rightarrow a-z\Delta \mid A-Z\Delta$
 - $\Delta \rightarrow a-z\Lambda \mid A-Z\Lambda \mid 0-9\Lambda \mid _ \Delta$
 - $\Lambda \rightarrow a-z \mid A-Z \mid 0-9 \mid a-z\Lambda \mid A-Z\Lambda \mid 0-9\Lambda \mid _ \Delta$

A1.3: Gleitkommazahlen in Programmiersprachen

- In Python kann man Gleitkommazahlen angeben durch $((0-9)^+.) + ((0-9)^+.(0-9)^+) + ((0-9)^+.(0-9)^+e(0-9)^+) + (.(0-9)^+) + (.(0-9)^+e(0-9)^+) + ((0-9)^+e(0-9)^+)$
 - DFA 5-Tupel $A = \{Q, \Sigma, \delta, q_0, F\}$
 - $Q = \{q_0, q_1, q_2\}$
 - $\Sigma = \{0-9, e, .\}$
 - $F = \{q_1, q_2\}$
 - $\delta :$



- Grammatik 4-Tupel $G_r = \{N, T, P, \Gamma\}$
 - $N = \{S, A, B\}$
 - $T = \{0-9, e, .\}$
 - $P :$
 - $S = 0S \mid 1S \mid 2S \mid 3S \mid 4S \mid 5S \mid 6S \mid 7S \mid 8S \mid 9S \mid .A \mid eB$
 - $A = 0A \mid 1A \mid 2A \mid 3A \mid 4A \mid 5A \mid 6A \mid 7A \mid 8A \mid 9A \mid eB$
 - $B = 0B \mid 1B \mid 2B \mid 3B \mid 4B \mid 5B \mid 6B \mid 7B \mid 8B \mid 9B \mid 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$
- In dem was ich herausfinden konnte gilt all dies auch für Java, mit dem einfachen Unterschied, dass Java noch den Suffix "f" für float erlaubt und Hexadezimal

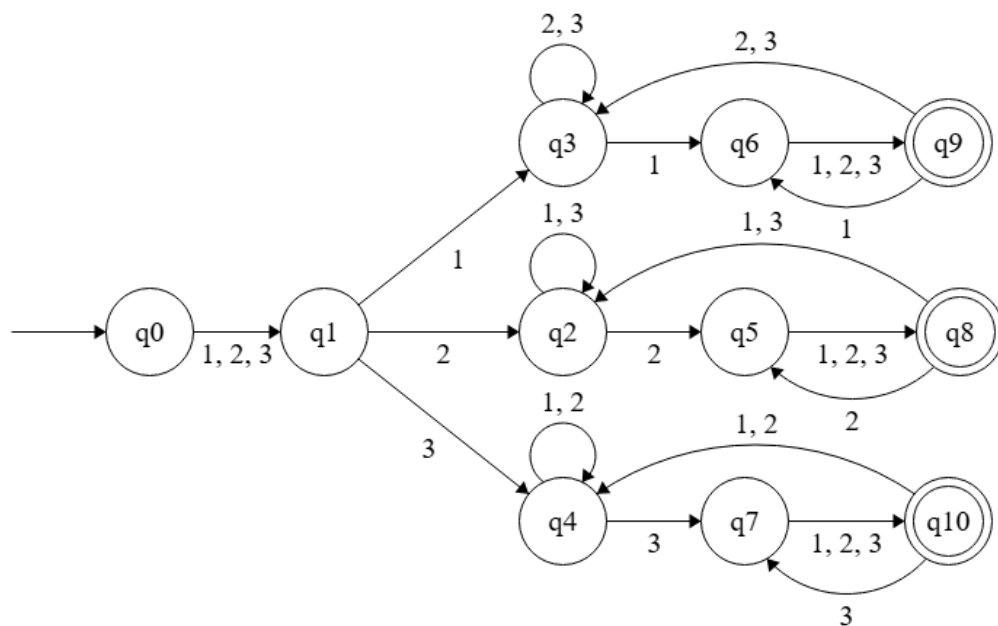
A1.4: Mailadressen?

- Der erste Ausdruck reicht nicht weil er einerseits nur Buchstaben akzeptiert, wobei normale Emailadressen auch Zahlen, Punkte, Bindestriche und Unterstriche enthalten können, und andererseits weil vor und nach dem Punkt nach diesem regex nur jeweils ein Buchstabe stehen könnte
- Verbesserung $\rightarrow (a + b + c + \dots + z + 0 + 1 + \dots + 9 + . + _ + -)^+ @ (a + b + c + \dots + z + 0 + 1 + \dots + 9 + . + _ + -)^+ . (a + b + c + \dots + z + 0 + 1 + \dots + 9 + . + _ + -)^+$
 - Anmerkung \rightarrow eigentlich müsste dieser jetzt noch durch die Bedingung keines Punktes oder Bindestrichs am Anfang oder Ende des Lokalteils

oder Domaintails sowie die Berücksichtigung des Zeichen Minimums und Maximums ergänzt werden, allerdings wäre der resultierende reguläre Ausdruck so lang und unübersichtlich, dass ich hier annehme dass dies nicht im Sinne der Aufgabe ist

A1.5: Der zweitletzte Buchstabe

- 5-Tupel $A = \{Q, \Sigma, \delta, q_0, F\}$
 - $Q = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8, q_9, q_{10}\}$
 - $\Sigma = \{1, 2, 3\}$
 - $F = \{q_8, q_9, q_{10}\}$
 - $\delta :$



A1.6: Sprache einer regulären Grammatik

- $L(G) = \{a((b+c)^*dc)^*(b+c)^*d(a+b) \mid (a,b,c,d) \in \Sigma\}$
- Ja, beispielsweise den DFA mit folgender Übergangsfunktion

