Theoretische Informatik Serie 4

Benjamin Simmonds, Dario Näpfer, Fabian Bösiger

Aufgabe 10

(a)

Wir führen zunächst einen Beweis durch Widerspruch und nehmen an, dass L_1 regulär ist. Es existiert also ein endlicher Automat M mit den Zuständen Q, der die Sprache L_1 akzeptiert.

Betrachten wir im Anschluss die Wörter $b, b^2, ..., b^m$, wobei $m = |Q| + 1$. Somit existieren mehr Wörter als M Zustände hat, dass heisst es existieren i, j mit $i \neq j$, so dass $\hat{\delta}(q_0, b^i) = \hat{\delta}(q_0, b^j)$. Nach Lemma 3.3 gilt also, dass $\forall z \in \{a, b\}^+ : b^i z \in L_1 \Leftrightarrow b^j z \in L_1$.

Für $z = ab^{(2i+1)}$ gilt $b^i z = b^i ab^{(2i+1)} \in L_1$, jedoch ist $b^j z = b^j ab^{(2i+1)} \notin L_1$, da $i \neq j$. Dies ist ein Widerspruch zur Aussage $\forall z \in \{a, b\}^+ : b^i z \in L_1 \Leftrightarrow b^j z \in L_1$, die Annahme ist somit falsch und es folgt, dass L_1 nicht regulär ist.

Die entsprechende direkte Argumentation ist, dass der Automat M, der die Sprache L_1 erkennt, eine Möglichkeit haben muss, sich das Wort ω zu merken, um anschliessend zu erkennen, wenn das Wort ein zweites Mal rückwärts und ein drittes Mal vorkommt. Da die Länge von ω beliebig gross werden kann, muss der Automat folgendermassen beliebig viele Zustände haben, was nicht möglich ist, da wir die Länge von ω immer $|\omega| = |Q| + 1$ wählen können. Für jeden Automaten, der L_1 erkennt, muss somit gelten, dass $\hat{\delta}(q_0, b^i) \neq \hat{\delta}(q_0, b^j)$, oder in anderen Worten $\forall z \in \{a, b\}^+ : b^i z \in L_1 \not\Leftrightarrow b^j z \in L_1$. Nach Lemma 3.3 kann der Automat M also kein endlicher Automat sein und die Sprache L_1 ist somit nicht regulär.

(b)

Wir nehmen an, L_2 sei regulär und führen einen Beweis durch Widerspruch.

Nach dem Pumping-Lemma existiert für L_2 eine Konstante $n_0 \in \mathbb{N}$, so dass sich jedes Wort $\omega \in \Sigma^*$ mit $|\omega| \geq n_0$ zerlegen lässt in $\omega = yxz$, wobei:

1. $|yx| \leq n_0$
2. $|x| \geq 1$
3. Entweder $\{yx^k z \mid k \in \mathbb{N}\} \subseteq L_2$ oder $\{yx^k z \mid k \in \mathbb{N}\} \cap L_2 = \emptyset$

Wir betrachten das Wort $\omega = a^{n_0} b^{n_0} c^{4n_0^2}$, das offensichtlich in L_2 enthalten ist. Wir teilen ω in die Teilwörter $\omega = yxz$. Da nach Aussage 1. des Pumping-Lemmas gilt, dass $|yx| \leq n_0$, und nach Aussage 2. $|x| \geq 1$, muss x für alle möglichen Zerlegungen mindestens ein a enthalten.

Da offensichtlich gilt, dass $\{yx^k z \mid k = 1 \in \mathbb{N}\} \cap L_2 \neq \emptyset$, muss nach Aussage 3. des Pumping-Lemmas gelten, dass $\{yx^k z \mid k \in \mathbb{N}\} \subseteq L_2$. Jedoch ist $yx^2 z = a^{n_0 - 1} a^2 b^{n_0} c^{4n_0^2} = a^{n_0+1} b^{n_0} c^{4n_0^2} \in \{yx^k z \mid k \in \mathbb{N}\}$ nicht in L_2 enthalten, da $n + 1 + n = 2n + 1 \not\leq \sqrt{4n^2}$. Somit ist dies ein Widerspruch zur Annahme, und es folgt, dass L_2 nicht regulär ist.

Aufgabe 11

(a)

Wir führen einen Beweis durch Widerspruch und nehmen an, dass $L_3 = \{0^{n!} \mid n \in \mathbb{N}\}$ regulär ist. Sei $y_{1,k} = 0^{(k+1)!-k!}$ das erste Wort in der Sprache $L_{k!} = \{y \mid 0^{k!}y \in L_3\}$.

Nach Satz 3.1 folgt, dass eine Konstante c existiert, die unabhängig von k ist, so dass die Kolmogorov-Komplexität $K(0^{(k+1)!-k!}) \leq \lceil log_2(1+1) \rceil + c = 1 + c := d$. Somit gilt $K(0^{(k+1)!-k!}) \leq d$ für alle $k \in \mathbb{N}$. Dies ist aber nicht möglich, weil die Anzahl aller Programme, deren Länge kleiner oder gleich d sind, höchstens 2^d und somit endlich ist, die Menge $\{0^{(k+1)!-k!} \mid k \in \mathbb{N}\}$ jedoch unendlich gross ist. Es kann nicht sein, dass unendlich viele Wörter eine endliche Kolmogorov-Komplexität besitzen. Aus diesem Widerspruch folgt, dass L_3 nicht regulär ist.

(b)

Wir führen einen Beweis durch Widerspruch unter der Verwendung von Lemma 3.3 und nehmen an, dass L_4 regulär ist, was bedeutet dass ein endlicher Automat M mit den Zuständen Q existiert, welcher L_4 akzeptiert.

Betrachten wir im Anschluss die Wörter $0, 1, 10, 11, ..., Bin(m)$, wobei $m = |Q| + 1$. Somit existieren mehr Wörter als M Zustände hat, dass heisst es existieren $0 \leq i, j \leq m$ mit $i \neq j$, so dass $\hat{\delta}(q_0, Bin(i)) = \hat{\delta}(q_0, Bin(j))$. Nach Lemma 3.3 gilt also, dass $\forall z \in \{0,1\}^+ : Bin(i)\#z \in L_4 \Leftrightarrow Bin(j)\#z \in L_4$.

Für $z = 0^i$ gilt $Bin(i)\#0^i \in L_4$, jedoch ist $Bin(j)\#0^i \notin L_4$, da $i \neq j$ und deshalb $Nummer(Bin(j)) = j \neq |0^i|$. Dies ist ein Widerspruch zur Aussage $\forall z \in \{0,1\}^+ : Bin(i)\#z \in L_4 \Leftrightarrow Bin(j)\#z \in L_4$, die Annahme ist somit falsch und es folgt, dass L_4 nicht regulär ist.

Aufgabe 12

Wir führen einen Beweis durch Widerspruch und nehmen an, dass $L = \{\omega_i \mid i \in \mathbb{N}\}$ regulär ist. Sei $y_{1,k} = 0^{|\omega_{k+1}|-|\omega_k|}$ das erste Wort in der Sprache $L_k = \{y \mid \omega_i y \in L\}$.

Nach Satz 3.1 folgt, dass eine Konstante c existiert, die unabhängig von k ist, so dass die Kolmogorov-Komplexität $K(0^{|\omega_{k+1}|-|\omega_k|}) \leq \lceil log_2(1+1) \rceil + c = 1 + c := d$. Somit gilt $K(0^{|\omega_{k+1}|-|\omega_k|}) \leq d$ für alle $k \in \mathbb{N}$. Die Anzahl aller Programme mit Länge d ist höchstens 2^d und somit endlich. Da gemäss Aufgabenstellung $|\omega_{k+1}| - |\omega_k| \geq log_2 log_2 k$, und $log_2 log_2 k$ streng monoton wachsend ist, folgt dass die Menge $\{0^{|\omega_{k+1}|-|\omega_k|} \mid k \in \mathbb{N}\}$ unendlich gross ist. Es kann aber nicht sein, dass unendlich viele Wörter eine endliche Kolmogorov-Komplexität besitzen. Aus diesem Widerspruch folgt, dass L nicht regulär sein kann.