AuB: 7.Übung (19.12.23)

Dienstag, 19. Dezember 2023 10:21

A-Äquivalenz und L-Äquivalenz (HA6)

s1.) *(Diese Aufgabe ist eine schriftliche Hausaufgabe, die bewertet wird.)*
Inspiriert durch die Definition der **A-Äquivalenz** eines beliebigen endlichen Automaten $A = (Q, \Sigma, \delta, q_0, F)$ definieren wir eine binäre Relation für eine beliebige formale Sprache L über Σ auf folgende Weise:

$u \sim_L v \quad \longleftrightarrow_{df} \quad \forall w \in \Sigma^* \ \ (uw \in L \leftrightarrow vw \in L)$

Diese Relation heißt **L-Äquivalenz**.

Beweisen Sie, dass $\sim_L$ tatsächlich eine Äquivalenzrelation über Σ^* ist!

S1.) L-Äquivalenz. gegeben sei $L \subseteq \Sigma^*$: $u \sim_L v \leftrightarrow_{df} \bigwedge_{w \in \Sigma^*} (uw \in L \leftrightarrow vw \in L)$

- reflexiv: Sei $u \in \Sigma^*$, dann gilt $\bigwedge_{w \in \Sigma^*} (uw = uw)$, für alle $u, v \in \Sigma^*$
  → Also $(uw \in L \leftrightarrow uw \in L)$, d.h. $u \sim_L u$
- symmetrisch: zu zeigen: $\bigwedge_{u \in \Sigma^*} \bigwedge_{v \in \Sigma^*} (u \sim_L v \to v \sim_L u)$
  Es sei $u \sim_L v$, d.h. $\bigwedge_{w \in \Sigma^*} (uw \in L \leftrightarrow vw \in L)$, d.h. $\bigwedge_{w \in \Sigma^*} (vw \in L \leftrightarrow uw \in L)$
  → Also ist $v \sim_L u$
- transitiv: zu zeigen: $\bigwedge_{x \in \Sigma^*} \bigwedge_{y \in \Sigma^*} \bigwedge_{z \in \Sigma^*} (x \sim_L y \wedge y \sim_L z \to x \sim_L z)$
  Es sei $x \sim_L y \wedge y \sim_L z$, z.z. $x \sim_L z$ / Wenn $xw \in L$, dann (wegen) $yw \in L$, Wenn $yw \in L$, dann (wegen) $zw \in L$
  ↳ Also, wenn $xw \in L$, dann $zw \in L$
  Es sei $w \in \Sigma^*$ umgekehrt / Wenn $zw \in L$, dann $yw \in L$ UND $yw \in L$, dann $xw \in L$
  ↳ Also, wenn $zw \in L$, dann $xw \in L$
  → Es gilt: $\bigwedge_{w \in \Sigma^*} (xw \in L \leftrightarrow zw \in L)$, d.h. $x \sim_L z$

s2.) *(Diese Aufgabe ist eine schriftliche Hausaufgabe, die bewertet wird.)*
Wir betrachten wieder die drei Sprachen über dem Alphabet $\Sigma = \{a, b\}$:

a) die Menge der Wörter, in denen die Anzahl der a's durch vier teilbar ist.
b) die Menge der Wörter, in denen die Zeichenkette *abba* vorkommt.
c) die Menge der Wörter, in denen kein Paar aufeinanderfolgender a's mehr vorkommt, sobald ein Paar aufeinanderfolgender b's vorgekommen ist.

Geben Sie für jede dieser Sprachen eine vollständige Beschreibung aller Äquivalenzklassen der *L-Äquivalenz* an, d.h. bestimmen Sie die zugehörige Faktormenge.!

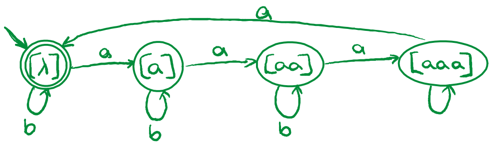
// Es gibt soviele Äquivalenzklassen, wie es Zustände im Automaten gibt

s.2a) $L_a = \{w \in \{a,b\}^* \mid |w|_a \equiv_4 0\}$

$[\lambda] = \{\lambda, \dots, \overset{b^k}{bbb}, \dots, \overset{4a}{aaaabbb}, \overset{\dots}{\dots}\ \overset{8a}{}\} = \{w \mid |w|_a \equiv_4 0\}$ 1
$[a] = \{a, \dots, abbb, \dots, aaaaa, \dots, aaaaabbb, \dots\} = \{w \mid |w|_a \equiv_4 1\}$ 2
$[aa] = \{aa, \dots, aabb, \dots, aaaaaabbb, \dots\} = \{w \mid |w|_a \equiv_4 2\}$ 3
$[aaa] = \{aaa, \dots, aaabb, \dots, aaaaaaabbb, \dots\} = \{w \mid |w|_a \equiv 3\}$ 4

$L_a/\sim_{L_a} = \{[\lambda], [a], [aa], [aaa]\}$ ↳ $\text{ind}(L_a) = 4$ (Index)

$L(A_a) = L_a$



// minimal

Zustandsfunktion: $f_A : \Sigma^* \longmapsto Q$
$f_A(w) = \delta^*(q_0, w)$

Zustandsklassen: $[q] = \{w \mid \delta^*(q_0, w) = q\}$ → Zerlegung von Σ^*

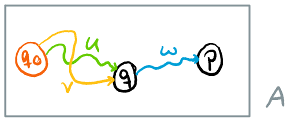
A-Äquivalenz mit folgender Eigenschaft:

- Lemma 1: $u \sim_A v \to \bigwedge_{w \in \Sigma^*} (uw \sim_A vw)$

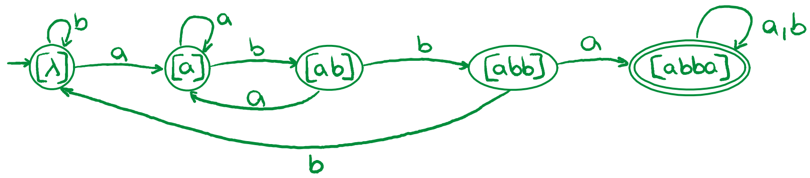
A-Äquivalenz mit folgender Eigenschaft:

· Lemma 1: $u \sim_A v \rightarrow \bigwedge_{w \in \Sigma^*} (uw \sim_A vw)$

insbesondere: $(uw \in L(A) \leftrightarrow vw \in L(A))$



S.2b) $L_B = \{w \in \{a,b\}^* \mid abba \subseteq w\}$



// A_b - minimal

$L(A_b) = L_b$

$L_b/_{\sim L_b} = \{[\lambda], [a], [ab], [abb], [abba]\}$ ↳ $ind(L_b) = 5$

S.2c) $\overline{L_c} = \{w \in \{a,b\}^* \mid \bigvee_x \bigvee_y \bigvee_z (w = xbbyaaz)\}$

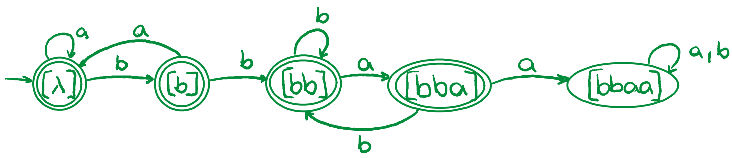
L_c-Äquivalenzklassen: [λ] ... alles ohne "bb" und am Ende kein "b"

[b] ... alles ohne "bb" und am Ende ein "b"

[bb] ... alles mit "bb" und kein "aa" und am Ende ein "b"

kleinste Wort in Klasse → [bba] .... alles mit "bb" und kein "aa" später und am Ende ein "a"

[bbaa] .... alles mit "bb" und ein "aa" später



// $L(A_c) = L_c$