AuB: 2.Hausaufgabe(08.11.23) - Cora Zeitler

Samstag, 4. November 2023 15:45

Alphabete, Wörter, Sprachen

s1.) *(Diese Aufgabe ist eine schriftliche Hausaufgabe, die bewertet wird.)*

a) Es seien $L_1, L_2, L_3 \subseteq \Sigma^*$ drei Sprachen über einem Alphabet Σ.
Beweisen Sie: $L_1 \cdot (L_2 \cap L_3) \subseteq L_1 \cdot L_2 \cap L_1 \cdot L_3$

b) Es seien L_1, L_2, L_3 Sprachen über dem Alphabet $\{|\}$.
Gilt $L_1(L_2 \cap L_3) = L_1 \cdot L_2 \cap L_1 \cdot L_3$?

c) Es seien $L_1 \subseteq \Sigma_1^*$ und $L_2, L_3 \subseteq \Sigma_2^*$ und $\Sigma_1 \cap \Sigma_2 = \emptyset$.
Gilt $L_1(L_2 \cap L_3) = L_1 \cdot L_2 \cap L_1 \cdot L_3$?

1b) $L_1 \cdot (L_2 \cap L_3) = L_1 \cdot L_2 \cap L_1 \cdot L_3$ (mit $L_1, L_2, L_3 \subseteq \Sigma^*$)

↳ die Gleichung gilt nicht. Beweis mit Gegenbeispiel

Es sei: $L_1 = \{|, ||\}$

$L_2 = \{|\}$

$L_3 = \{||\}$

$L_1 \cdot (L_2 \cap L_3) \overset{?}{=} L_1 \cdot L_2 \cap L_1 \cdot L_3$

$L_1 \cdot \emptyset = \{||, |||\} \cap \{|||, ||||\}$

$\emptyset = \{|||\}$

↯ $\emptyset = \{|||\}$ ist eine falsche Aussage,
d.h wie oben Beh., die Gleichung gilt nicht.

6/6

1a) zu zeigen: jedes Wort in $\{L_1 \cdot (L_2 \cap L_3)\}$ ist auch in $\{L_1 \cdot L_2 \cap L_1 \cdot L_3\}$ enthalten

angenommen: $w \in \{L_1 \cdot (L_2 \cap L_3)\}$

das heißt: es gibt ein Wort x in (L_1) und ein Wort y in $(L_2 \cap L_3)$ → d.h. y ist sowohl in L_2 als L_3 enthalten

↳ somit gibt es ein $w = x \cdot y$

wir wissen: $y \in \{L_2 \cap L_3\} \Leftrightarrow y \in \{L_2\}$ und $y \in \{L_3\}$

weiter zu zeigen: $y \in \{L_1 \cdot L_2 \cap L_1 \cap L_3\}$

das heißt es gibt: $z_1 \in \{L_1 \cdot L_2\}$ und $z_2 \in \{L_1 \cdot L_3\}$

↳ dann gilt: $y = z_1 = z_2$ ↯ → Das gilt eben nicht unbedingt. Die Richtung ist nämlich falsch!

L ist schon eine Menge, die Mengeklammer brauchst du hier also nicht.

somit ist $w = x \cdot y = x \cdot z_1 = x \cdot z_2$

das heißt es gibt ein w in $\{L_1 \cdot L_2 \cap L_1 \cdot L_3\}$, weil sich w in beiden Teilmengen befindet ∎

3/6

1c) $L_1 \cdot (L_2 \cap L_3) = L_1 \cdot L_2 \cap L_1 \cdot L_3$ (mit $L_1 \subseteq \Sigma_1^*$ und $L_2, L_3 \subseteq \Sigma_2^*$ und $\Sigma_1 \cap \Sigma_2 = \emptyset$

Ich werde erst 2 Bsp. zur Verständlichkeit nutzen

① es sei $L_1 = \{z\}$ // $\Sigma_1^* =$ nur z

$L_2 = \{b, bb\}$, $L_3 = \{b\}$ // $\Sigma_2^* =$ nur b

zu①: $L_1 \cdot (L_2 \cap L_3) = L_1 \cdot L_2 \cap L_1 \cdot L_3$

$\{z\} \cdot \{b\} = \{zb, zbb\} \cap \{zb\}$

$\{zb\} = \{zb\}$

② es sei: $L_1 = \{z\}$, $L_2 = \{bbb\}$, $L_3 = \{b, bb\}$ // Beschreibung analog zu ①

zu②: $L_1 \cdot (L_2 \cap L_3) = L_1 \cdot L_2 \cap L_1 \cdot L_3$

$\{z\} \cdot \emptyset = \{zbbb\} \cap \{zb, zbb\}$

$\emptyset = \emptyset$

→ Σ_1^* und Σ_2^* sind disjunkte Alphabete, d.h. es gibt keine gemeinsamen Buchstaben

→ d.h. die Wörter w in L_1 und die Wörter w in L_2 und L_3 sind unabhängig voneinander

→ bei $L_1 \cdot (L_2 \cap L_3)$ wählen wir zuerst Wörter w aus L_1 aus und nehmen dann den Schnitt von L_2 und L_3

↳ $w \in L_1$ kann nicht beeinflusst werden von $w \in L_2, L_3$

↳ wenn es keinen Schnitt bei L_2 und L_3 gibt ist es automatisch $\emptyset$ auf beiden Seiten der Gleichung (②)

↳ wenn es einen Schnitt bei L_2 und L_3 gibt, dann ist der Buchstabe in L_2 und L_3 und dann gibt es das Wort auf der linken und rechten Seite der Gleichung (①)

↳ wenn es keinen Schnitt bei L_2 und L_3 gibt ist es automatisch $\emptyset$ auf beiden Seiten der Gleichung (②)

↳ wenn es einen Schnitt bei L_2 und L_3 gibt, dann ist der Buchstabe in L_2 und L_3 und dann gibt es das Wort auf der Linken und Rechten Seite der Gleichung (①)

↳ beide Fälle wären erfüllt

→ somit ist $L_1 \cdot (L_2 \cap L_3)$ äquivalent zu $L_1 \cdot L_2 \cap L_1 \cdot L_3$, da die Auswahl von Wörtern in L_1 und die Auswahl von Wörtern in L_2 und L_3 unabhängig voneinander erfolgen ∎ ✓

8/8 :)

s2.) *(Diese Aufgabe ist eine schriftliche Hausaufgabe, die bewertet wird.)*

Für diese Aufgabe legen wir das Alphabet $\Sigma = \{a, b\}$ zu Grunde.
Beweisen Sie induktiv:
Für jedes Wort $w \in \Sigma$ gilt: $aw \neq wb$.

2) Induktionsanfang: für Wörter w der Länge 0 gilt:

$|w| = 0 \Rightarrow w = \lambda$

$aw \neq wb \Leftrightarrow a\lambda \neq \lambda b \Leftrightarrow a \neq b$ ✓ ✓

für Wörter w der Länge 1 gilt

$|w| = 1$: 1. Fall: $aa \neq ab$

2. Fall: $ab \neq bb$ ✓ ✓

Ind. vorrausetzung: für Wörter w der Länge k gilt:

$|w| = k$: $aw \neq wb$ ✓

Ind. behauptung: für Wörter w der Länge $k+1$ gilt:

$|w| = k+1$: $aw \neq wb$

also $w = \boxed{ax = xa = bx = xb}$ (① ② ③ ④) mit $x \in \Sigma^*$, wobei $|x| = n$ ✓

$ax = bx$ ist natürlich nicht richtig!

Ind. beweis: ① $aw \neq wb \Leftrightarrow a(\underbrace{ax}_{k+1}) \neq (\underbrace{ax}_{k+1})b$ ✓, egal wie man versucht Buchstaben auszugleichen, man hat links immer ein a zu viel und rechts ein b im Überfluss (✓)

→ $\underbrace{aaax}_{k+2} \neq \underbrace{aaxb}_{k+2}$ ...., d.h. das es bleibt immer ungleich

② $aw \neq wb \Leftrightarrow axa \neq axb$, egal was man für x einsetzt, die letze Stelle des Wortes bleibt immer ungleich ✓

③ $aw \neq wb \Leftrightarrow abx \neq bxb$, egal was man für x einsetzt, die erste Stelle des Wortes bleibt immer ungleich ✓ (✓)

④ $aw \neq wb \Leftrightarrow axb \neq xbb$, egal was man für x einsetzt um das Wort auszugleichen es gibt links immer ein a mehr

→ $aaxb \neq xabb$ ↘ Bsp analog wie ①

→ allgemeiner zum Beweis lässt sich auch darstellen wie folgt:

- Es sei $x = wu$ mit $u \in \Sigma$ wobei $|w| = k$ und $|x| = k+1$
- dann gilt:
  ① $ax = a(wu) = (aw)u = (wa)u$
  ② $xa = (wu)a = (wa)u = (aw)u$
  ③ $bx = b(wu) = (bw)u = (wb)u$
  ④ $xb = (wu)b = (wb)u = (bw)u$

  } unter Verwendung Assoziativgesetz f

Warum soll $aw = wa$ gelten?

- mit Ind. vorrausetzung gilt: $aw \neq wb \Leftrightarrow (aw)u \neq (wb)u$ ✓

→ u ist selbe Buchstabe, d.h. egal was man für u einsetzt, (aw) und (wb) bleiben, d.h die Gesamtwörter sind unterschiedlich ∎

6/8

An einigen Stellen etwas unsauber, aber der richtige Ansatz :)

Sorry für die späte Korrektur, ich versuche mich zu bessern.

Σ: 23/28 :)