Institut für Informatik **Ludwig-Maximilians-Universität München**
Prof. Dr. Volker Heun

Formale Sprachen und Komplexität, SS 18
Übungsblatt 10

Abgabe: bis Mo 09.07.2018 8 Uhr

Nach Bearbeitung dieses Übungsblattes sollten Sie:

	Check
Gut mit den Begriffen *entscheidbar* und *semi-entscheidbar* umgehen können.	
Mit der Reduktion von formalen Sprachen argumentieren können.	

Diese Ziele sind wichtige Hinweise für die Klausur!

Aufgabe 10-1 schriftlich bearbeiten

Reduktion von formalen Sprachen: Eine Trockenübung II

Sei $\Sigma = \{a, b\}$ und $A = \left\{ w \in \Sigma^* \;\middle|\; |w|_a = |w|_b \right\}$. Sei $\Gamma = \{c, d, e\}$ und $C = \left\{ c^n d^n e^n \;\middle|\; n \in \mathbb{N} \right\}$.

a) Zeigen Sie, dass A auf C reduzierbar ist, das heißt $A \leq C$.

Lösungsvorschlag:

Definiere
$$\begin{array}{rccl} f: & \Sigma^* & \to & \Gamma^* \\ & w & \mapsto & c^{|w|_a}\, d^{|w|_b}\, e^{|w|_b} \end{array}$$

Beispiele:
$$\begin{array}{rll} baba \in A \subseteq \Sigma^* & f(baba) = ccddee \in C & \subseteq \Gamma^* \\ aab \in \overline{A} \subseteq \Sigma^* & f(aab) \;=\; ccde \in \overline{C} & \subseteq \Gamma^* \end{array}$$

f ist total, also für jedes $w \in \Sigma^*$ definiert (insbesondere ist $f(\varepsilon) = \varepsilon$)

f ist berechenbar, beispielsweise durch das folgende SML-Programm.

Sei $w \in \Sigma^*$ beliebig, sei $i = |w|_a$ und $j = |w|_b$. Dann ist $f(w) = c^i d^j e^j$.
Damit gilt: $w \in A$ gdw. $i = j$ gdw. $f(w) \in C$.

Insgesamt gilt also $A \leq C$.

```
fun f(w) =
    let
        fun anzahl(x, nil)   = 0
          | anzahl(x, y::L)  = (if x=y then 1 else 0) + anzahl(x,L)

        fun hoch(x, 0)       = nil
          | hoch(x, n)       = x :: (hoch(x, n-1))

        val L  = String.explode(w)       (* string -> char list *)

        val L' = hoch(#"c", anzahl(#"a", L))
                 @
                 hoch(#"d", anzahl(#"b", L))
                 @
                 hoch(#"e", anzahl(#"b", L))

        val w' = String.implode(L')      (* char list -> string *)
    in
        w'
    end
```

b) Es gibt ein Entscheidungsverfahren für C (etwa eine kleine Modifikation einer früher behandelten Turingmaschine). Was folgt daraus für die (Semi-)Entscheidbarkeit von A?

Bemerkung: *Das Ergebnis bringt natürlich keinen großen Erkenntnisgewinn über diese beiden entscheidbaren Sprachen. Zweck der Aufgabe ist es, die Technik der Reduktion an einem einfachen Beispiel zu illustrieren.*

Lösungsvorschlag:

Mit einer Variante einer früher betrachteten Turingmaschine kann man C entscheiden:
Für jedes $z \in \Gamma^*$ terminiert die Turingmaschine mit Bandinhalt 1 falls $z \in C$
und Bandinhalt 0 falls $z \notin C$.

Diese Turingmaschine kann jetzt auch benutzt werden, um A zu entscheiden:
Jedes $w \in \Sigma^*$ wird mit f auf ein $z \in \Gamma^*$ abgebildet,
mit dem man die Turingmaschine startet.
Das Ergebnis der Turingmaschine für z ist dann auch das richtige Ergebnis für w.

Für diese einfache Sprache A wäre die Entscheidbarkeit allerdings
auch mit anderen Mitteln leicht zu zeigen.

Aufgabe 10-2 (semi-)entscheidbar

Sei Σ ein Alphabet und sei $A \subseteq \Sigma^*$ eine formale Sprache.

a) A ist semi-entscheidbar gdw. *Bedingung*
Geben Sie möglichst viele Bedingungen an, für die das stimmt.

Lösungsvorschlag:

A ist semi-entscheidbar gdw.

- die halbe charakteristische Funktion χ'_A ist (Turing/WHILE/...)-berechenbar (Def)
- es gibt einen Algorithmus mit Eingabe $w \in \Sigma^*$,
  der für jedes $w \in A$ terminiert und 1 (oder *true* oder *ja*) liefert
  und für jedes $w \notin A$ nicht terminiert
- A ist rekursiv aufzählbar
- entweder $A = \emptyset$ oder es gibt eine totale und berechenbare Funktion $f : \mathbb{N} \to \Sigma^*$ mit $A = \{f(n) \mid n \in \mathbb{N}\}$
- es gibt eine Turingmaschine M mit $A = T(M)$
- A ist (mind.) vom Typ 0
- es gibt eine Grammatik G mit $A = L(G)$
- es gibt eine semi-entscheidbare Sprache B mit $A \leq B$
- A ist reduzierbar auf eine rekursiv aufzählbare Sprache
- A ist reduzierbar auf eine Sprache, die eine der genannten Bedingungen erfüllt
- ...

b) A ist entscheidbar gdw. *Bedingung*
Geben Sie möglichst viele Bedingungen an, für die das stimmt.

Lösungsvorschlag:

A ist entscheidbar gdw.

- die charakteristische Funktion χ_A ist (Turing/WHILE/...)-berechenbar (Def)
- es gibt einen Algorithmus mit Eingabe $w \in \Sigma^*$,
  der für jedes $w \in \Sigma^*$ terminiert
  und 1 (oder *true* oder *ja*) liefert falls $w \in A$
  und 0 (oder *false* oder *nein*) liefert falls $w \notin A$
- sowohl A als auch $\overline{A}$ sind semi-entscheidbar
- die halben charakteristischen Funktionen χ'_A und $\chi'_{\overline{A}}$
  sind beide (Turing/WHILE/...)-berechenbar
- sowohl A als auch $\overline{A}$ sind rekursiv aufzählbar
- entweder $A = \emptyset$ und $\overline{A} = \Sigma^*$
  oder $A = \Sigma^*$ und $\overline{A} = \emptyset$
  oder es gibt totale und berechenbare Funktionen
  $f : \mathbb{N} \to \Sigma^*$ mit $A = \{f(n) \mid n \in \mathbb{N}\}$
  $\overline{f} : \mathbb{N} \to \Sigma^*$ mit $\overline{A} = \{\overline{f}(n) \mid n \in \mathbb{N}\}$
- es gibt Turingmaschinen M und $\overline{M}$ mit $A = T(M)$ und $\overline{A} = T(\overline{M})$
- sowohl A als auch $\overline{A}$ sind (mind.) vom Typ 0
- es gibt Grammatiken G und $\overline{G}$ mit $A = L(G)$ und $\overline{A} = L(\overline{G})$
- es gibt eine entscheidbare Sprache B mit $A \leq B$
- es gibt semi-entscheidbare Sprachen B und C mit $A \leq B$ und $\overline{A} \leq C$
- ...

Aufgabe 10-3 a), b), c) und h) schriftlich bearbeiten

Post'sches Korrespondenzproblem

Jede der Teilaufgaben a) bis g) behandelt eine Instanz des Post'schen Korrespondenzproblems. Geben Sie zu jeder Instanz entweder eine Lösung an oder einen Beweis, dass keine Lösung existiert.

a) **schriftlich bearbeiten**

$x_1 = b \quad x_2 = ba \quad x_3 = ab$

$y_1 = ab \quad y_2 = abb \quad y_3 = abb$

Lösungsvorschlag:

Indexfolge:	keine Lösung
Wort:	
Begründung:	Für alle i ist $\lvert x_i\rvert < \lvert y_i\rvert$, also ist für jede Indexfolge das „obere“ Wort kürzer als das „untere“.

b) **schriftlich bearbeiten**

$x_1 = ca \quad x_2 = b \quad x_3 = aba$

$y_1 = bac \quad y_2 = ab \quad y_3 = a$

Lösungsvorschlag:

Indexfolge:	$3, 1, 2$
Wort:	$abacab$

c) **schriftlich bearbeiten**

$x_1 = bab \quad x_2 = ba \quad x_3 = ab$

$y_1 = bb \quad y_2 = abb \quad y_3 = ba$

Lösungsvorschlag:

Indexfolge:	keine Lösung
Wort:	
Begründung:	Keines der drei Paare kann Anfang einer Lösung sein.

d) $x_1 = aaaaa \quad x_2 = aa$

$y_1 = a \quad y_2 = aaaaa$

Lösungsvorschlag:

Indexfolge:	$1, 1, 1, 2, 2, 2, 2$
Wort:	$aaaaaaaaaaaaaaaaaaaaaaa = a^{23}$
Begründung:	Es kommt nur auf die Länge der erzeugten Wörter an, die Reihenfolge der Indizes spielt überhaupt keine Rolle. Eine Lösung, in der n_1-Mal Paar 1 vorkommt und n_2-Mal Paar 2, muss folgende Gleichung erfüllen: $5 \cdot n_1 + 2 \cdot n_2 = 1 \cdot n_1 + 5 \cdot n_2$, also $4 \cdot n_1 - 3 \cdot n_2 = 0$. Lösung: $n_1 = 3$ und $n_2 = 4$, also 3 Mal Paar 1 und 4 Mal Paar 2, Reihenfolge beliebig. Ob eine Gleichung der Form $c_1 \cdot n_1 + ... + c_k \cdot n_k = 0$ eine Lösung in den natürlichen Zahlen hat, ist entscheidbar. Deshalb ist das Post'sche Korrespondenzproblem über einelementigen Alphabeten entscheidbar.

e) $x_1 = abc \quad x_2 = abba \quad x_3 = c \quad x_4 = bbba \quad x_5 = abcc$
$y_1 = ab \quad y_2 = aabb \quad y_3 = ccc \quad y_4 = cbbb \quad y_5 = aab$

Lösungsvorschlag:

Indexfolge:	$1, 4, 5, 3$
Wort:	*abcbbbaabccc*

f) $x_1 = ba \quad x_2 = abb \quad x_3 = bab$
$y_1 = bab \quad y_2 = bb \quad y_3 = abb$

Lösungsvorschlag:

Indexfolge:	keine Lösung
Wort:	
Begründung:	Weder Paar 2 noch Paar 3 kann Anfang einer Lösung sein. Also muss jede Lösung mit Paar 1 anfangen. Dieser Anfang hat die Eigenschaft, dass das untere Wort gleich dem oberen Wort mit einem zusätzlichen b am Ende ist. Eine Indexfolge mit dieser Eigenschaft kann nur mit Paar 3 fortgesetzt werden, wodurch eine Indexfolge mit der selben Eigenschaft entsteht. In allen Indexfolgen, die Anfang einer Lösung sein könnten, ist somit das untere Wort länger als das obere.

g) $x_1 = b \quad x_2 = a \quad x_3 = ca \quad x_4 = abc$
$y_1 = ca \quad y_2 = ab \quad y_3 = a \quad y_4 = c$

Lösungsvorschlag:

Indexfolge:	$2, 1, 3, 2, 4$
Wort:	*abcaaabc*

h) **schriftlich bearbeiten**.
In den Fällen, in denen keine Lösung existiert, kann man die Unlösbarkeit also beweisen. Andererseits ist bewiesen, dass das Post'sche Korrespondenzproblem nicht entscheidbar ist. Warum ist das kein Widerspruch?

Lösungsvorschlag:

Wir haben nur einzelne Instanzen analysiert. Daraus folgt nicht unbedingt, dass unsere Analyse auf jede Instanz übertragbar ist.

Einige der obigen Beweise enthalten hinreichende Bedingungen für die Unlösbarkeit (zum Beispiel: wenn für keines der gegebenen Paare das eine Wort ein Präfix des anderen ist, dann gibt es keine Lösung).

Das Unentscheidbarkeitsergebnis schließt solche hinreichenden Bedingungen nicht aus, besagt aber, dass notwendige Bedingungen für die Unlösbarkeit nicht existieren, jedenfalls keine entscheidbaren.

Mit anderen Worten: Man kann noch so viele hinreichende Bedingungen für die Unlösbarkeit finden, es wird immer Instanzen geben, die keine dieser Bedingungen erfüllen und trotzdem unlösbar sind.

Aufgabe 10-4 Modifiziertes Post'sches Korrespondenzproblem

Über einem beliebigen Alphabet Σ sind das Postsche Korrespondenzproblem und das Modifizierte Postsche Korrespondenzproblem gegeben durch:

PCP : $\{((x_1, y_1), (x_2, y_2), \ldots, (x_k, y_k)) | \exists i_1, i_2 \ldots i_n \in \{1, 2, \ldots, k\} : x_{i_1}, x_{i_2}, \ldots, x_{i_n} = y_{i_1}, y_{i_2}, \ldots y_{i_n}\}$

MPCP : $\{((x_1, y_1), (x_2, y_2), \ldots, (x_k, y_k)) | \exists i_2, i_3 \ldots i_n \in \{1, 2, \ldots, k\} : x_1, x_{i_2}, \ldots, x_{i_n} = y_1, y_{i_2}, \ldots y_{i_n}\}$

Zeigen Sie: $MPCP \leq PCP$.

Tipp: Die vermittelnde Funktion zwischen $MPCP$ und PCP ist gegeben durch:

$$\begin{array}{lcl} F : ((\Sigma^*)^2)^k & \to & ((\Sigma^*)^2)^{k+2} \\ \quad ((x_1, y_1), (x_2, y_2), \ldots, (x_k, y_k)) & \mapsto & ((\bar{x_1}, \grave{y_1}), (\acute{x_1}, \grave{y_1})(\acute{x_2}, \grave{y_2}), \ldots, (\acute{x_k}, \grave{y_k}), (\$, \#\$)) \end{array}$$

Wobei # und \$ Zeichen sind die nicht in Σ vorkommen, und für ein Wort $\omega = a_1 \ldots a_n$ $\bar{\omega}$ $\grave{\omega}$ $\acute{\omega}$ gegeben sind durch:

$$\bar{\omega} = \#a_1\#a_2 \ldots \#a_n\#, \quad \acute{\omega} = a_1\#a_2 \ldots \#a_n\#, \quad \grave{\omega} = \#a_1\#a_2 \ldots \#a_n$$

Lösungsvorschlag:

F ist total und berechenbar (was der Vollständigkeit halber noch zu zeigen wäre).
Sei K eine Instanz des modifizierten Post'schen Korrespondenzproblems.
Es ist jetzt zu zeigen : $K \in MPCP \Leftrightarrow F(K) \in PCP$

'$\Rightarrow$' Angenommen K hat eine Lösung $I = (1, i_2, \ldots, i_n)$. Wir müssen jetzt eine Lösung J von $F(K)$ finden. Wir müssen mit 1 beginnen, denn nur $(\bar{x_1}, \grave{y_1})$ beginnen beide mit einem #. $J = (1, ?, ?, ? \ldots ?)$
Die letzte Stelle der Lösung muss $(k + 2)$ sein, denn nur \$ und #\$ hören auf \$ auf. $J = (1, ?, ?, \ldots, k + 2)$
Jetzt zum Rest der Lösung. Wir wissen, dass x_{i_2} und y_{i_2} an x_1 und y_1 angeschlossen werden können. In $F(K)$ haben wir aber nur $\grave{x}_{i_2}$ und $\acute{y}_{i_2}$ zur Verfügung. Das macht aber nichts, die passen zu $(\bar{x_1}, \grave{y_1})$. Also $J = (1, i_2 + 1, ?, ?, \ldots, k + 2)$.
Die anderen Stellen von J überlegt man sich genauso: $J = (1, i_2{+}1, i_3{+}1, \ldots, i_k{+}1, k{+}2)$.
Also ist $F(K) \in PCP$

'$\Leftarrow$' Sei jetzt $F(K) \in PCP$, und $J = (j_1, \ldots, j_n)$ eine Lösung von $F(K)$. Wir suchen eine Lösung $I = (1, i_2, \ldots, i_n - 2)$ von K. Wieder wie gerade eben muss $j_1 = 1$ und $j_n = k + 2$ sein. Es passen also $(\bar{x_1}, \grave{y_1})$ zusammen. Dann passen auch (x_1, y_1) zusammen. $I = (1, ?, ?, \ldots ?)$
Weiter passt $\grave{x}_{j_2}$ zu $\acute{y}_{j_2}$ und schließt sich an die Lösung von $F(K)$ an. Also passt x_{j_2} zu y_{j_2}. Die finden wir an der Stelle $j_2 - 1$ in K. $I = (1, j_2 - 1, ?, ?, \ldots, ?)$.
So weiter überlegt man sich $I = (1, j_2 - 1, j_3 - 1, \ldots, j_{n-1} - 1)$