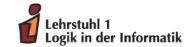
# ÜBUNGEN ZUR VORLESUNG GRUNDBEGRIFFE DER THEORETISCHEN INFORMATIK



THOMAS SCHWENTICK

GAETANO GECK, LUTZ OETTERSHAGEN, CHRISTOPHER SPINRATH, MARCO WILHELM



SOSE 2018 ÜBUNGSBLATT 10 19.06.2018

Abgabe bis spätestens am Dienstag, 26.06.2018,

- (vor der Vorlesung) im HG II, HS 3, oder
- in die Briefkästen im Durchgangsflur, der die 1. Etage der OH 12 mit dem Erdgeschoss der OH 14 verbindet.

Beachten Sie die Schließzeiten der Gebäude!

Ansonsten gelten die Hinweise von Blatt 1.

#### Aufgabe 10.1 [Kodierung einer Turingmaschine]

2 Punkte

Die Turingmaschine  $M=(\{s,p\},\{\triangleright,\sqcup,0,1\},\delta,s)$  über dem Alphabet  $\Sigma=\{0,1\}$  ist ein binärer Dekrementierer: Die Binärzahl  $w^R$  wird um 1 dekrementiert, falls  $\operatorname{Str2N}(w^R)>0$  gilt. Die Transitionen  $(q,\sigma,q',\sigma',d)$  aus  $\delta$  sind dabei wie folgt definiert:

q	$\sigma$	q'	$\sigma'$	d
$\overline{s}$	$\triangleright$	s	$\triangleright$	$\rightarrow$
s	0	s	0	$\rightarrow$
s	1	p	0	←
s	Ш	h	Ш	$\downarrow$
p	0	p	1	$\leftarrow$
p	$\triangleright$	h	$\triangleright$	$\downarrow$

a) Wie werden die Symbole und Zustände von M gemäß den im Abschnitt "Universelle Turingmaschinen" von Kapitel 16 genannten Konventionen kodiert, wenn M Eingabe für eine universelle Turingmaschine ist? Verwenden Sie die folgende Nummerierung der Zustände, Richtungen und Zeichen und ergänzen Sie die Tabellen:

x	$\operatorname{num}(x)$	$\mathrm{enc}(x)$
s	1	
h	4	
p	5	
<b>←</b>	1	
$\downarrow$	2	
$\rightarrow$	3	

<i>x</i>		$\mathrm{enc}(x)$
$\triangleright$	1	
Ш	2	
0	3	
1	4	

(1 Punkt)

b) Geben Sie die Kodierungen enc(s,1) und enc(p,0) an.

(1 Punkt)

#### Aufgabe 10.2 [Abschlusseigenschaften]

4 Punkte

(2 Punkte)

Es seien  $L_1$  und  $L_2$  Sprachen über einem Alphabet  $\Sigma$  und  $M_1$  bzw.  $M_2$  Turingmaschinen für die  $L_1 = L(M_1)$  und  $L_2 = L(M_2)$  gilt. Beschreiben Sie anschaulich, wie sich Turingmaschinen  $M_a$  und  $M_b$ für die folgende Sprachen  $L_a$  und  $L_b$  konstruieren lassen, so dass jeweils  $L_x = L(M_x)$  gilt. Bedenken Sie, dass  $M_x$  die Sprache  $L_x$  nicht entscheiden muss. Es ist also möglich, dass sie für einige Eingaben nicht terminiert. M1 und M2 erhalten Übergänge, die für alle verwerfenden oder nicht existenten Knoten

zum Anfang des Strings laufen und dann die andere TM durchlaufen.

**b**)  $L_b = L_1 \circ L_2$ (2 Punkte) erst M1 ab halten wieder and den Anfange des Strings, dann M2

#### Hinweis

a)  $L_a = L_1 \cup L_2$ 

 $M_1$  und  $M_2$  können als 1-String-Turingmaschinen angenommen werden. Für  $M_a$  und  $M_b$  dürfen Sie Mehrstring-Turingmaschinen verwenden. Sie müssen die (Mehrstring-) Turingmaschinen  $M_a$  und  $M_b$ weder formal aufschreiben, noch als Diagramm angeben. Beschreiben Sie jedoch die Arbeitsweise hinreichend verständlich und nutzen Sie dafür die Tatsache, dass Turingmaschinen beliebige andere Turingmaschinen simulieren können.

# Aufgabe 10.3 [Entscheidbarkeit, Semi-Entscheidbarkeit, Unentscheidbarkeit]

7 Punkte

Geben Sie für die folgenden Probleme an, ob sie jeweils

(muss ja nicht enden)

- entscheidbar,
- nicht entscheidbar aber semi-entscheidbar oder
- unentscheidbar

sind. Beweisen Sie Ihre Behauptungen. Gehen Sie davon aus, dass jede Turingmaschine die Zeichen 0 und 1 im Eingabe- und im Arbeitsalphabet enthält.

nicht entscheidbar -> Red analog zu a) Problem:

hello world

Gegeben: Turingmaschine M und  $k \in \mathbb{N}_0$ semi entscheidbar -> wie in tut

nicht entscheidbar -> Red (3 Punkte) Erzeugt M bei der Eingabe  $0^k$  die Ausgabe 1? Frage:

hello world **b)** Problem:

nicht semientscheidbar

Turingmaschine MGegeben: Frage:

-> unendlich viele eingaben müssten Erzeugt M bei keiner Eingabe die Ausgabe 1?

## Hinweis

Laut den Konventionen im Abschnitt "Universelle Turingmaschinen" von Kapitel 16 ist die Kodierung der Zeichen 0 und 1 eindeutig gegeben durch num(0) = 3 und num(1) = 4.

## Aufgabe 10.4 [Primitiv Rekursive Funktionen]

2 Punkte

Zeigen Sie: Die Funktion even $(x): \mathbb{N}_0 \to \mathbb{N}_0$  mit

$$even(x) = \begin{cases} 1, & x \text{ ist gerade} & \mathbf{g} \\ 0, & x \text{ ist ungerade} & \mathbf{h} \end{cases}$$

ist primitiv rekursiv.

16,F18+ Konstante Funktionen sind prim. rek. g,h konstant

Zusammensetzungen aus prim. rek. Funktionen sind prim. rek. primitive rekursion:

f(0) = 0

f(x+1) = even(x)

even(0) = 1

even(x+1) = f(x)

Version vom 21.06.2018 - 10:38 Uhr