

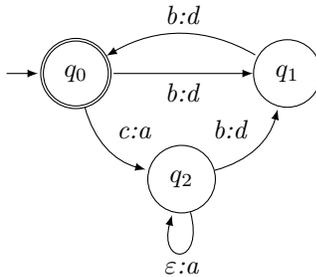
Einführung in die Computerlinguistik

Hausaufgabe 4 (FST), Abgabe 19.05.2014

Laura Kallmeyer

SS 2014, Heinrich-Heine-Universität Düsseldorf

Aufgabe 1 Betrachten Sie folgenden FST:



- Auf welche Strings bildet der FST die folgenden Eingaben ab (es können mehrere Strings sein, eventuell auch unendlich viele, die Abbildung ist nicht immer eindeutig):
 (a) bb (b) cbb (c) ε (d) $bbcbcbbbb$
- Welche Strings akzeptiert dieser FST und wie transformiert er sie? (Eine Beschreibung der Transformation in Worten genügt.)

Lösung:

- Angabe der entsprechenden regulären Ausdrücke, da es teilweise unendlich viele Bilder gibt:
 (a) dd (b) a^+dd (c) ε (d) dda^+dda^+dddd
- Akzeptiert wird die von $((c|\varepsilon)bb)^*$ denotierte reguläre Sprache. Ausgabe ist jeweils ein Wort, in dem jedes c durch eine beliebig lange Folge von a 's (mindestens eins) ersetzt ist, und jedes b im Eingabewort durch ein d .

Aufgabe 2

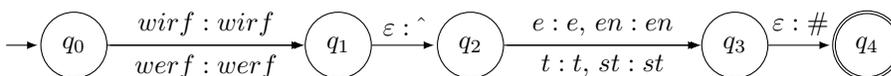
Erstellen Sie Finite State Transducer, die folgende Formen von "werfen" erkennen und deren Zwischenebene bzw. deren Analyse ausgeben:

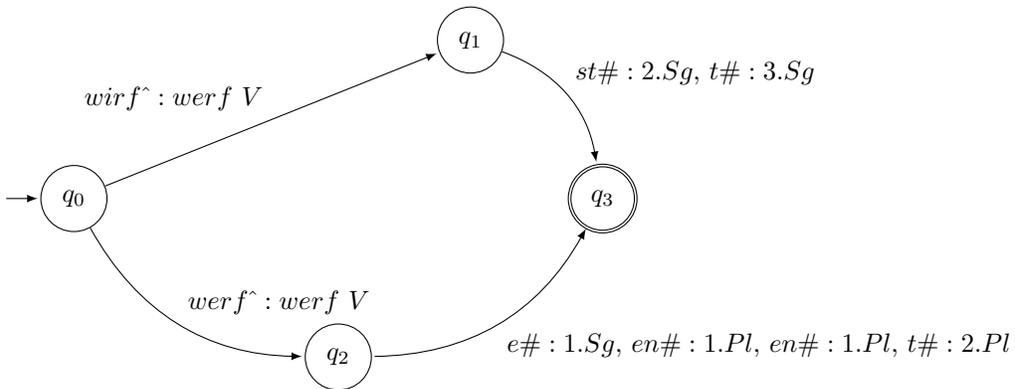
Wortform	Zwischenebene	Analyse	Wortform	Zwischenebene	Analyse
werfe	$werf \hat{e} \#$	$werf V 1.Sg$	werfen	$werf \hat{en} \#$	$werf V 1.Pl$
wirfst	$wirf \hat{st} \#$	$werf V 2.Sg$	werft	$werf \hat{t} \#$	$werf V 2.Pl$
wirft	$wirf \hat{t} \#$	$werf V 3.Sg$	werfen	$werf \hat{en} \#$	$werf V 3.Pl$

Zunächst sollen in einem ersten FST die Symbole $\hat{}$ und $\#$ für Morphem- und Wortgrenze eingefügt werden, ein zweiter FST macht dann die morph. Analyse. (Beide FSTs können für Generierung natürlich auch in der anderen Richtung betrieben werden.)

Lösung:

FST, der Morphemgrenzen markiert:





Aufgabe 3 Geben Sie einen FST an, der die von $(a^*bcd a^*)^*$ denotierte Sprache erzeugt, und der 1. Die a s nicht-deterministisch entweder löscht oder durch e ersetzt, 2. jede Folge bc durch ein f ersetzt und 3. alle d s löscht.

Lösung:

