

Hochschule Bremerhaven
University of Applied Sciences

Fakultät II – Management und Informationssysteme
Informatik
Modul Theoretische Informatik
Prof. Dr.-Ing Henrik Lipskoch

Protokoll zu Aufgabenblatt 09: Team: ti2023_22

Von

Ekane Njoh Junior Lesage

Matrikelnmr: 40128

Aguiwo II Steve

Matrikelnmer: 40088

Inhalt

I.	Aufgabe 1	3
a.	Wortwahl.....	3
b.	Turingmaschine	3
c.	Buchstabe Angabe.....	Fehler! Textmarke nicht definiert.
d.	Startsymbol	Fehler! Textmarke nicht definiert.
e.	Beispiel aus Aufgabe 01	Fehler! Textmarke nicht definiert.
II.	Aufgabe 02	5
III.	Aufgabe 03	5
IV.	Literaturverzeichnis	5

I. Aufgabe 1

Bei dieser Aufgabe handelt es sich darum eine Turingmaschine zu definieren, die ein einziges Wort aus unserer Sprache vom Blatt 05 Aufgabe 02 erkennen kann. Dabei sollen wir einiges beachten:

- Die Länge des Wortes soll wenigstens 10 Buchstaben sein.
- Das Wort soll den Wiederholungsteil Ihrer Sprache enthalten (der Teil, warum die Sprache echt kontextfrei ist)
- Bei Erfolg soll die Maschine nach links laufen und den Vornamen eines der Teammitglieder auf das Band schreiben; alle anderen Buchstaben werden zu blank.

a. Wortwahl

Es soll ein Wort gewählt werden, dass mindestens 10 Buchstaben lang ist. Hierfür haben wir folgendes Wort ausgewählt, das 12 Zeichen lang ist:

if [10 – eq 8]; then echo ok ; fi

b. Turingmaschine

Zusammen haben wir folgende Turingmaschine M definiert:

$$M = (Z, \Sigma, \Gamma, \delta, z_0, \square, E)$$

$$Z = \{ z_0; z_1; \dots; z_n; z_s; z_r \}$$

z_s ist hierbei der Endzustand, in dem die Maschine den Namen „JUNIOR“ auf das Band schreibt.

z_r stellt der Ablehnungszustand, in den die Maschine übergeht, wenn das Wort nicht den Anforderungen entspricht.

$$\Sigma = \{ 'if'; '['; '10'; '-eq'; '8'; ']' ; ';' ; 'then'; 'echo'; 'ok'; ' '; ' '; 'fi' \}$$

$$\Gamma = \{ 'if', '[', '10', '-eq', '8', ']', ';', 'then', 'echo', 'ok', 'fi', '\square', 'J', 'U', 'N', 'I', 'O', 'R', 'Ö', '\%', '\$', '\&', '\beta', '\ddot{A}' \}$$

$$E = \{ z_s \}$$

Das Konzept für die Zustandsüberföhrungsfunktion δ , beinhaltet spezifische Zustände für jede Phase der Wortanalyse, beginnend mit dem Startzustand, der das Lesen des Wortes initiiert. Jeder Zustand ist zuständig für die Überprüfung bestimmter Wortteile und Strukturen. Die Maschine wechselt Zustände basierend auf der erfolgreichen Erkennung dieser Teile und führt entsprechende Aktionen aus, wie das Schreiben von Markierungen oder die Bewegung des Lese-/Schreibkopfes. Bei Erfüllung aller Bedingungen schreibt die Maschine "JUNIOR" rückwirkend auf dem Band.

$$\begin{aligned}
& \delta(z_0, if) = (z_1, if, R) & \delta(z_8, echo) = (z_9, echo, R) \\
& \forall x \in \Gamma \setminus \{if\} : \delta(z_0, x) = (z_r, x, R) & \forall x \in \Gamma \setminus \{echo\} : \delta(z_8, x) = (z_r, x, R) \\
& \delta(z_1, []) = (z_2, [], R) & \delta(z_9, ok) = (z_{10}, ok, R) \\
& \forall x \in \Gamma \setminus \{[\} : \delta(z_1, x) = (z_1, if, R) & \forall x \in \Gamma \setminus \{ok\} : \delta(z_9, x) = (z_r, x, R) \\
& \delta(z_2, 10) = (z_3, 10, R) & \delta(z_{10}, ;) = (z_{11}, :, R) \\
& \forall x \in \Gamma \setminus \{10\} : \delta(z_2, x) = (z_r, x, R) & \forall x \in \Gamma \setminus \{;\} : \delta(z_{10}, x) = (z_r, x, R) \\
& \delta(z_3, -ge) = (z_4, -ge, R) & \delta(z_{11}, fi) = (z_{12}, fi, R) \\
& \forall x \in \Gamma \setminus \{-ge\} : \delta(z_3, x) = (z_r, x, R) & \delta(z_{12}, \square) = (z_{13}, \square, L) \\
& \delta(z_4, 8) = (z_5, 8, R) & \delta(z_r, \ddot{O}) = (z_r, R, L) \\
& \forall x \in \Gamma \setminus \{8\} : \delta(z_4, x) = (z_r, x, R) & \delta(z_r, \ddot{A}) = (z_r, O, L) \\
& \delta(z_5,]) = (z_6,], R) & \delta(z_r, \beta) = (z_r, I, L) \\
& \forall x \in \Gamma \setminus \{]\} : \delta(z_5, x) = (z_r, x, R) & \delta(z_r, \&) = (z_r, N, L) \\
& \delta(z_6, ;) = (z_7, :, R) & \delta(z_r, \$) = (z_r, U, L) \\
& \forall x \in \Gamma \setminus \{;\} : \delta(z_6, x) = (z_r, x, R) & \delta(z_r, \% \$ \& \beta \ddot{A}) = (z_r, J, L) \\
& \delta(z_7, then) = (z_8, then, R) & \forall x \in \Sigma : \delta(z_r, x) = (z_r, \square, L) \\
& \forall x \in \Gamma \setminus \{then\} : \delta(z_7, x) = (z_r, x, R) & \delta(z_r, \square) = (z_r, \square, R)
\end{aligned}$$

Aufgrund der Aufforderung der Aufgabe für die Wortakzeptanz der Maschine (ein einziges Wort), haben wir die Maschine sehr eingeschränkt gestaltet.

II. Aufgabe 02

Konfigurationsüberleitungen für unser Wort

III. Aufgabe 03

IV. Literaturverzeichnis