Hochschule Bremerhaven

University of Applied Sciences

Fakultät II – Management und Informationssysteme

Informatik

Modul Theoretische Informatik

Prof. Dr.-Ing Henrik Lipskoch

**Protokoll zu Aufgabenblatt 08: Team: ti2023\_22**

**Von**

**Ekane Njoh Junior Lesage**  Matrikelnmr: 40128

**Aguiwo II Steve** Matrikelnmer: 40088

Inhalt

[I. Aufgabe 1 2](#_Toc153489035)

[a. Auswahl der Sprache 2](#_Toc153489036)

[b. Definition der Grammatik 2](#_Toc153489037)

[c. Umwandlung in der Kuroda-Normalform 3](#_Toc153489038)

[d. Wortbildung 4](#_Toc153489039)

[e. Nachweis 4](#_Toc153489040)

[f. Erste Aufteilung 5](#_Toc153489041)

[g. Zweite Aufteilung 5](#_Toc153489042)

[h. Literaturverzeichnis 6](#_Toc153489043)

## Aufgabe 1

Bei dieser Aufgabe handelt es sich darum, uns eine Sprache auszusuchen bzw. zu erfinden, in der KNF auszudrücken und anschließen nachzuweisen, dass es eine Typ-1 Sprache ist, indem das Pummping-Lemma für kontextfreie Sprachen angewendet wird. Das heißt

## Auswahl der Sprache

Hier haben wir uns nach langer Überlegung entschieden, uns die folgende Sprache auszusuchen:

Unsere Sprache könnte als kontextsensitiv betrachtet, weil wir eine Regel hinzugefügt haben, die die Anzahl der Symbole in einer spezifischen Reihenfolge einschränkt. Die Regel i>j>k definiert, dass die Anzahl der ‚a‘s größer sein soll, als die Anzahl der ‚b’s, die auch größer sein soll als die Anzahl der ‚c’s.

## Definition der Grammatik

eine mögliche Grammatik für unsere Sprache würde aus Folgendem bestehen:

* Ein Alphabet:
* Eine Variablenmenge:
* Produktionsregeln:
* Und ein Startsymbol

## Umwandlung in der Kuroda-Normalform

Aus der Folie 8-3 wissen wir, dass eine Typ-1 Sprache in der KNF akzeptabel ist, wenn sie nur Regeln der Form: aufweist.

Unsere Regeln entsprechend nicht alle diesem Format, daher müssen sie noch angepasst bzw. erweitert werden.

Beginnen wir mit der 1. Regel:

Damit haben wir unsere erste Regel erweitert. Da wir die 2. Regel nicht umwandeln müssen, weil die schon die KNF entspricht, machen wir mit der 3. Regeln weiter:

Diese Regel muss geändert werden, weil links eine Variable gefolgt von einem Buchstaben und rechts eine Variable gefolgt von einem Buchstaben steht. Dies erfolgt, indem wir c eine Variable zuweisen.

Die 5. Regel: muss nicht angepasst werden, weil die schon in der KNF ist.

Wir sind nun soweit, dass wir unsere umgewandelte Grammatik aufstellen können. Das Ergebnis ist also Folgendes:

Wir erhalten somit eine neue Variablenmenge:

Das Alphabet und das Startsymbol bleiben dabei unberührt. Jetzt ist nur nachzuweisen, dass unsere Sprache des Typen 1 ist.

## Wortbildung

Wenden Wir unsere Produktionsregeln an damit wir ein Wort zur Anwendung des Pummping-Lemmas verwenden können.

Das erzeugte Wort ist also wobei . Es respektiert also die Produktionsregeln.

## Nachweis

Zum Nachweisen der Typisierung unserer Sprache wenden wir das Pummping-Lemma für kontextfreie Sprachen an und nur wenn wir dabei scheitern, heißt es unsere Sprache ist tatsächlich vom Typ-1. Angenommen wird dann, dass unsere Sprache L kontextfrei ist.

Wir wissen aus Folie 6-1, dass eine Sprache kontextfrei ist genau dann, wenn,

:

lässt sich zerlegen in mit



Unser Wort ist dann wir haben bereits die Entscheidbarkeit unserer Sprache in Bezug auf dieses Wort nachgewiesen.

Wir wählen . Dies gilt für die unsere zwei verschiedenen Zerlegungen für das gleiche Wort.

## Erste Aufteilung

Wir prüfen nun ob, die Bedingungen für das Pummping-Lemma erfüllt sind:

Alle erforderlichen Bedingungen haben wir erfüllt. Das Abpumpen kann anfangen.

Für erhalten wir

Für erhalten wir

Für erhalten wir *b*b

Für erhalten wir

Für erhalten wir

Für ist das Pummping-Lemma verletzt und das Wort gehört nicht mehr zur Sprache, weil es die Bedingung nicht mehr erfüllt, dass . Es gibt dort gleich oder mehr ‚b’s als ‚a’s, was unzulässig ist.

## Zweite Aufteilung

Wir prüfen nun ob, die Bedingungen für das Pummping-Lemma erfüllt sind:

Alle erforderlichen Bedingungen haben wir erfüllt. Das Abpumpen kann anfangen.

Für erhalten wir

Für erhalten wir

Für erhalten wir

Für erhalten wir

Für erhalten wir

Für scheitert das Pummping-Lemma, weil es gleich oder mehr ‚b’s als ‚a’s gibt.

Es ist noch zu beweisen, ob das Pummping-Lemma für andere Aufteilungen gelten könnte, aber dafür müssten erheblich viele unterschiedliche Kombinationen getestet werden.

Bis dahin verbleiben wir mit der Schlussfolgerung, dass unsere Sprache kontextsensitiv und somit vom Typ-1 in der Chomsky-Hierarchie ist.

## Neue Aufteilung

Zu zeigen, dass unsere Sprache echt-kontextsensitiv ist, müssen wir auch ein Wort aus dieser Sprache finden, für das das Pummping-Lemma gilt. Dafür wenden wir unsere Produktionsregeln an, um das Wort zu bilden.

Das Wort ist dann .

Wir wählen

Wir prüfen nun ob, die Bedingungen für das Pummping-Lemma erfüllt sind:

Alle erforderlichen Bedingungen haben wir erfüllt. Das Abpumpen kann anfangen.

Für erhalten wir

Für erhalten wir

Für erhalten wir

Für erhalten wir

Für erhalten wir

Für ist das Pummping-Lemma verletzt und das Wort gehört nicht mehr zur Sprache, weil es die Bedingung nicht mehr erfüllt, dass . Es gibt dort keinen c, was unzulässig ist.

## Literaturverzeichnis

[1]<https://de.wikipedia.org/wiki/Chomsky-Hierarchie>

Letzter Zugriff am 10.12.2023 um 02:08 Uhr

[2] <https://elli.hs-bremerhaven.de/goto.php?target=file_338582_download>

Letzter zugriff am 10.12.2023 um 03:02 Uhr

[3] <https://elli.hs-bremerhaven.de/goto.php?target=file_336977_download>

Letzter Zugriff am 110.12.2023 um 02:30 Uhr