Informatik III - Tutorium XIII & XV (SR -107)

Tut Nr. 12 – Üb11, kontextfreie Sprachen

David Münch

Universität Karlsruhe (TH)
Institut für Informatik
ITI Wagner

30. Januar 2008

Universität Karlsruhe (TH)
Forschungsuniversität • gegründet 1825

Inhaltsverzeichnis

Inhaltsverzeichnis

Inhaltsverzeichnis

Inhaltsverzeichnis

Organisatorisches

Email: muenchdavid@gmail.com

https://www.stud.uni-karlsruhe.de/~uhbro/

Tutorium 13: Mittwochs 8:00 Uhr - Raum -107
Tutorium 15: Mittwochs 9:45 Uhr - Raum -107

Übungsblattabgabe Donnerstag.

Was wollen wir heute erreichen?

Was wollen wir heute erreichen?

- Pumping-/Odgen's-Lemma für kfG anwenden

Was wollen wir heute erreichen?

- Pumping-/Odgen's-Lemma für kfG anwenden
- Algorithmus für nutzlose Symbole

Aufgabe 7

Gegeben sei die Grammatik $G_1 = (\Sigma, V, S, R)$ mit $\Sigma = \{a, b\}$, $V = \{S, A, B\}$ und der Regelmenge R:

$$\begin{aligned} S &\rightarrow aA \mid \varepsilon \\ A &\rightarrow aS \mid bB \\ B &\rightarrow bB \mid bA \,. \end{aligned}$$

1. Welchen Chomsky-Typ hat G_1?
2. Konstruiere einen nichtdeterministischen endlichen Automaten für $L(G)$ mithilfe des Verfahrens aus dem Skript.

Aufgabe 7

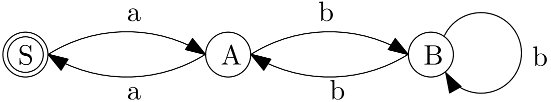
1. G_1 ist eine CH-3-Grammatik.

Aufgabe 7

1. G_1 ist eine CH-3-Grammatik.
2. Wir konstruieren einen NEA $\mathcal{N} = (Q, \Sigma, \delta, q_0, F)$ mit

$$\begin{aligned} Q &= V = \{S, A, B\} \\ q_0 &= S \\ F &= \{A \in V \mid (A \to \varepsilon) \in R\} = \{S\} \text{ und} \\ \delta(A, a) &= \{B \mid (A \to aB) \in R\} \end{aligned}$$

Die Übergangsfunktion δ ist gegeben durch:



Aufgabe 8

Gegeben sei die Grammatik $G_2 = (\Sigma, V, S, R)$ mit $\Sigma = \{a, c, d\}$, $V = \{A, B, C, D, S\}$ und folgender Regelmenge R:

$$
\begin{array}{rcl}
S & \rightarrow & A \mid aB \mid aC \\
A & \rightarrow & B \mid C \mid cAd \\
B & \rightarrow & S \mid Ba \mid \varepsilon \\
C & \rightarrow & D \mid c \\
D & \rightarrow & d \mid dDD
\end{array}
$$

1. Welchen Chomsky-Typ hat G_2?
2. Überführe G_2 in Chomsky-Normalform.

Aufgabe 8

1. G_1 ist eine CH-2-Grammatik.

Aufgabe 8

1. G_1 ist eine CH-2-Grammatik.
2. 1. Schritt:
   Alle Regeln enthalten auf der rechten Seite nur Symbole aus V oder nur ein Symbol aus Σ.

$$
\begin{aligned}
S &\rightarrow A \mid Y_aB \mid Y_aC \\
A &\rightarrow B \mid C \mid Y_cAY_d \\
B &\rightarrow S \mid BY_a \mid \varepsilon \\
C &\rightarrow D \mid Y_c \\
D &\rightarrow Y_d \mid Y_dDD \\
Y_a &\rightarrow a \\
Y_c &\rightarrow c \\
Y_d &\rightarrow d
\end{aligned}
$$

Aufgabe 8

2. Schritt:
Alle rechten Seiten haben Länge ≤ 2.

$$\begin{array}{rcl}
S & \rightarrow & A \mid Y_a B \mid Y_a C \\
A & \rightarrow & B \mid C \mid Y_c C_1 \\
B & \rightarrow & S \mid B Y_a \mid \varepsilon \\
C & \rightarrow & D \mid Y_c \\
D & \rightarrow & Y_d \mid Y_d C_2 \\
C_1 & \rightarrow & A Y_d \\
C_2 & \rightarrow & D D \\
Y_a & \rightarrow & a \\
Y_c & \rightarrow & c \\
Y_d & \rightarrow & d
\end{array}$$

Aufgabe 8

3. Schritt:
Es kommen keine Regeln $A \to \varepsilon$ vor. Berechne $V' = \{A \in V \mid A \xrightarrow{*} \varepsilon\} = \{A, B, S\}$. Wir streichen die Regel $B \to \varepsilon$ und führen als neue Regeln $S \to Y_a, B \to Y_a, C_1 \to Y_d$ ein.

$$
\begin{array}{rcl}
S & \to & A \mid Y_aB \mid Y_aC \mid Y_a \\
A & \to & B \mid C \mid Y_cC_1 \\
B & \to & S \mid BY_a \mid Y_a \\
C & \to & D \mid Y_c \\
D & \to & Y_d \mid Y_dC_2 \\
C_1 & \to & AY_d \mid Y_d \\
C_2 & \to & DD \\
Y_x & \to & x \quad //x = a, b, c
\end{array}
$$

Aufgabe 8

4. Schritt:
Ersetzung aller Kettenregeln $A \rightarrow B$.
Es gibt nur einen Kreis: $S \rightarrow A \rightarrow B \rightarrow S$ mit beteiligten Regeln $S \rightarrow A, A \rightarrow B, B \rightarrow S$.
Diese Regeln werden gelöscht und alle Vorkommen von S, A, B in allen Regeln werden durch S ersetzt.
Wir erhalten die Regelmenge:

Aufgabe 8

$$
\begin{aligned}
S &\rightarrow S \mid Y_a S \mid Y_a C \mid Y_a \\
S &\rightarrow S \mid C \mid Y_c C_1 \\
S &\rightarrow S \mid S Y_a \mid Y_a \\
C &\rightarrow D \mid Y_c \\
D &\rightarrow Y_d \mid Y_d C_2 \\
C_1 &\rightarrow S Y_d \mid Y_d \\
C_2 &\rightarrow D D \\
Y_a &\rightarrow a \\
Y_c &\rightarrow c \\
Y_d &\rightarrow d
\end{aligned}
$$

Aufgabe 8

Nach Löschen aller Regeln der Form $S \to S$ erhalten wir dann:

$$
\begin{aligned}
S &\to Y_aS \mid Y_aC \mid Y_a \mid C \mid Y_cC_1 \mid SY_a \\
C &\to D \mid Y_c \\
D &\to Y_d \mid Y_dC_2 \\
C_1 &\to SY_d \mid Y_d \\
C_2 &\to DD \\
Y_a &\to a \\
Y_c &\to c \\
Y_d &\to d
\end{aligned}
$$

Aufgabe 8

Topologisches Sortieren der verbleibenden an Kettenregeln beteiligten Variablen liefert z.B. die Reihenfolge der Variablen $S, C, Y_a, D, C_1, Y_c, Y_d$. Wir verarbeiten die Regeln in umgekehrter Reihenfolge und ersetzen Kettenregeln.

Aufgabe 8

$$
\begin{aligned}
S &\rightarrow Y_aS \mid Y_aC \mid a \mid d \mid Y_dC_2 \mid c \mid Y_cC_1 \mid SY_a \\
C &\rightarrow d \mid Y_dC_2 \mid c \\
D &\rightarrow d \mid Y_dC_2 \\
C_1 &\rightarrow SY_d \mid d \\
C_2 &\rightarrow DD \\
Y_a &\rightarrow a \\
Y_c &\rightarrow c \\
Y_d &\rightarrow d
\end{aligned}
$$

Diese Regelmenge ist in Chomsky-Normalform.

Aufgabe 8

Zum Schluss müssen wir die Grammatik noch ergänzen durch die Regeln $S' \to \varepsilon$ und $S' \to S$ für ein neues Startsymbol S', da die ursprüngliche Grammatik die Ableitung $S \xrightarrow{*} \varepsilon$ zulässt.

Aufgabe 9

Gegeben sei die Grammatik $G_3 = (\Sigma, V, S, R)$ mit $\Sigma = \{a, b\}$ und $V = \{A, B, S, X\}$. Die Regelmenge R sei gegeben durch:

$$
\begin{aligned}
S &\rightarrow AX \mid AB \\
X &\rightarrow SB \\
A &\rightarrow a \\
B &\rightarrow b
\end{aligned}
$$

1. Lässt sich der CYK-Algorithmus auf G_3 anwenden? Begründe Deine Antwort.
2. Prüfe mithilfe des CYK-Algorithmus, ob das Wort $aabb$ in $L(G)$ liegt, nachdem G gegebenenfalls so angepasst wurde, dass der Algorithmus anwendbar ist.

Aufgabe 9

1. Der CYK-Algorithmus lässt sich auf G_3 anwenden, da G_3 eine kontextfreie Grammatik in Chomsky-Normalform ist.

Aufgabe 9

1. Der CYK-Algorithmus lässt sich auf G_3 anwenden, da G_3 eine kontextfreie Grammatik in Chomsky-Normalform ist.
2. Die Grammatik muss also nicht modifiziert werden. Die Mengen V_{ij} berechnen sich wie folgt:

$$
\begin{array}{llll}
V_{11} = \{A\} & V_{12} = \emptyset & V_{13} = \emptyset & V_{14} = \{S\} \\
V_{22} = \{A\} & V_{23} = \{S\} & V_{24} = \{X\} & \\
V_{33} = \{B\} & V_{34} = \emptyset & & \\
V_{44} = \{B\} & & &
\end{array}
$$

V_{24} etwa enthält X, weil G_3 die Regel $X \to SB$ enthält und $S \in V_{23}$ sowie $B \in V_{44}$ gilt. Da V_{14} S enthält, ist das Wort in $L(G_3)$.

Aufgabe 10

Gegeben sei die Grammatik $G_4 = (\Sigma, V, S, R)$ mit $\Sigma = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, +, -, \cdot, /\}$ und $V = \{S, Z\}$. R sei durch die folgenden Ableitungsregeln in Chomsky-Normalform gegeben:

$$
\begin{aligned}
S &\rightarrow S + S \mid S - S \mid S \cdot S \mid S/S \mid Z \\
Z &\rightarrow 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9 \mid 0Z \mid 1Z \mid 2Z \mid 3Z \mid 4Z \mid 5Z \mid 6Z
\end{aligned}
$$

a)
Bestimme einen Ableitungsbaum des Wortes $100 - 23 + 57/4$. Ist die Grammatik eindeutig oder inhärent mehrdeutig? Begründe deine Antwort.

Aufgabe 10

b)
Welchen Chomsky-Typ hat G_4, welchen $L(G_4)$? Gib eine entsprechende Grammatik für $L(G_4)$ an, falls sich die Chomsky-Typen von $L(G_4)$ und G_4 unterscheiden.
c)
Welchen Chomsky-Typ hat $L(G_4)$, wenn die erste Regel in R durch die Regel

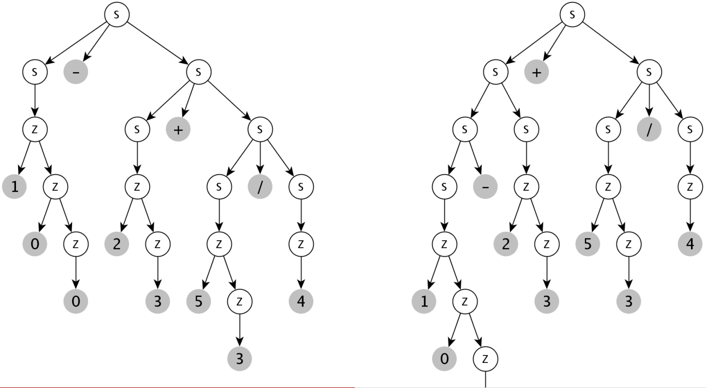
$$S \to (S + S) \mid (S - S) \mid S \cdot S \mid S/S \mid Z$$

ersetzt wird? Begründe Deine Antwort.

Augabe 10

a)

G_4 ist inhärent mehrdeutig, da es mehrere Syntaxbäume für das gegebene Wort gibt .



Augabe 10

b)
G_4 ist eine Chomsky-2-Grammatik. $L(G_4)$ ist allerdings eine reguläre Sprache, die auch durch die folgende (rechtslineare) Chomsky-3-Grammatik beschrieben wird.

$$\begin{aligned} S &\rightarrow 0Z \mid 1Z \mid \cdots \mid 9Z \mid 0S \mid 1S \mid \cdots \mid 9S \\ Z &\rightarrow +S \mid -S \mid \cdot S \mid /S \mid 0E \mid \cdots \mid 9E \\ E &\rightarrow \varepsilon \end{aligned}$$

Augabe 10

c)

Nach Ersetzung der ersten Regel ist $L(G_4)$ nicht mehr regulär, da sich Ausdrücke mit korrekter Klammerstruktur wie z.B. $((3+4)-8)\cdot(1+2)$ bilden lassen. Die Sprache der korrekten Klammerausdrücke ist nicht regulär (siehe Übung). Die regulären Sprachen sind aber genau die Sprachen, für die es eine Chomsky-3-Grammatik gibt. Daher kann es für $L(G_4)$ mit der modifizierten Grammatik keine Chomsky-3-Grammatik geben. Da $L(G_4)$ durch eine Chomsky-2-Grammatik beschrieben wird, ist $L(G_4)$ nach Modifikation von G_4 also kontextfrei.

Definition: Pumping-Lemma für kontextfreie Sprachen

Für jede kontextfreie Sprache L gibt es eine Konstante $n \in \mathbb{N}$, so dass sich jedes Wort $z \in L$ mit $|z| \geq n$ so als $z = uvwxy$ schreiben lässt, dass $|vx| \geq 1$, $|vwx| \leq n$ und für alle $i \geq 0$ das Wort $uv^iwx^iy \in L$ ist.

Beispiel

Zeige mithilfe des Pumping-Lemmas, dass $L = \{a^i b^i c^i | i \geq 1\}$ nicht kontextfrei ist.

Aufgabe 1

Zeige mithilfe des Pumping-Lemmas, dass $L = \{a^i b^j c^i d^j | i, j \geq 1\}$ nicht kontextfrei ist.

Definition: Ogden's Lemma

Für jede kontextfreie Sprache L gibt es eine Konstante $n \in \mathbb{N}$, so dass für jedes Wort $z \in L$ mit $|z| \geq n$ gilt: Wenn wir in z mindestens n Buchstaben markieren, so lässt sich z so als $z = uvwxy$ schreiben, dass von den mindestens n markierten Buchstaben mindestens einer zu vx gehört und höchstens n zu vwx gehören und für alle $i \geq 0$ das Wort $uv^iwx^iy \in L$ ist.

Beispiel

Zeige, dass $L = \{a^i b^j a^k | i \neq 0 \Rightarrow j = max\{i, k\}\}$ nicht kontextfrei ist.

Aufgabe 2

Zeige mithilfe von Ogden's Lemma, dass
$L = \{w0^iw | w \in \{0,1\}^*, 0 \leq i\}$ nicht kontextfrei ist.

Definition: Nutzlose Variablen

Sei G eine kontextfreie Grammatik. Eine Variable A heisst nutzlos, falls es keine Ableitung $S \xrightarrow{*} w$ gibt, $w \in \Sigma^*$, in der A vorkommt.

Aufgabe 3

Enthält die folgende Grammatik $G = (V, \Sigma, S, R)$ mit
$V = \{A, B, C, S\}$, $\Sigma = \{a, b\}$ und $R = \{$

$$\begin{array}{lcl} S & \to & AB|CA, \\ A & \to & a, \\ B & \to & BC|AB, \\ C & \to & aB|b \end{array}$$

$\}$
nutzlose Symbole?

Wie erkennen wir nutzlose Variablen?

Wie erkennen wir nutzlose Variablen?

1. Berechnung von $V' \subseteq V$ mit $A \in V'$ genau dann wenn es $w \in \Sigma$ gibt mit $A \xrightarrow{*} w$.

Wie erkennen wir nutzlose Variablen?

1. Berechnung von $V' \subseteq V$ mit $A \in V'$ genau dann wenn es $w \in \Sigma$ gibt mit $A \xrightarrow{*} w$.
   - Erstelle $V' = \emptyset$ und eine Queue $Q = \emptyset$.

Wie erkennen wir nutzlose Variablen?

1. Berechnung von $V' \subseteq V$ mit $A \in V'$ genau dann wenn es $w \in \Sigma$ gibt mit $A \xrightarrow{*} w$.
   - Erstelle $V' = \emptyset$ und eine Queue $Q = \emptyset$.
   - Für alle $A \in V$ mit $A \rightarrow w$, $w \in \Sigma^*$, setze $Q = V' = V' \cup \{A\}$

Wie erkennen wir nutzlose Variablen?

1. Berechnung von $V' \subseteq V$ mit $A \in V'$ genau dann wenn es $w \in \Sigma$ gibt mit $A \xrightarrow{*} w$.
   - Erstelle $V' = \emptyset$ und eine Queue $Q = \emptyset$.
   - Für alle $A \in V$ mit $A \to w$, $w \in \Sigma^*$, setze $Q = V' = V' \cup \{A\}$
   - Für alle $A \in Q$:
     - Ersetze $B \to \alpha A \beta$ mit $B \to \alpha w \beta$ durch Anwenden der Regel $A \to w$; α, $\beta \in (V \cup \Sigma)^*$, $w \in \Sigma^*$.

Wie erkennen wir nutzlose Variablen?

1. Berechnung von $V' \subseteq V$ mit $A \in V'$ genau dann wenn es $w \in \Sigma$ gibt mit $A \xrightarrow{*} w$.
   - Erstelle $V' = \emptyset$ und eine Queue $Q = \emptyset$.
   - Für alle $A \in V$ mit $A \rightarrow w$, $w \in \Sigma^*$, setze $Q = V' = V' \cup \{A\}$
   - Für alle $A \in Q$:
     - Ersetze $B \rightarrow \alpha A \beta$ mit $B \rightarrow \alpha w \beta$ durch Anwenden der Regel $A \rightarrow w$; $\alpha, \beta \in (V \cup \Sigma)^*$, $w \in \Sigma^*$.
     - Exisitert nun $B \rightarrow w'$ mit $w' \in \Sigma^*$, füge B in Q und V' ein.

Wie erkennen wir nutzlose Variablen?

1. Berechnung von $V' \subseteq V$ mit $A \in V'$ genau dann wenn es $w \in \Sigma$ gibt mit $A \xrightarrow{*} w$.
   - Erstelle $V' = \emptyset$ und eine Queue $Q = \emptyset$.
   - Für alle $A \in V$ mit $A \rightarrow w$, $w \in \Sigma^*$, setze $Q = V' = V' \cup \{A\}$
   - Für alle $A \in Q$:
     - Ersetze $B \rightarrow \alpha A \beta$ mit $B \rightarrow \alpha w \beta$ durch Anwenden der Regel $A \rightarrow w$; α, $\beta \in (V \cup \Sigma)^*$, $w \in \Sigma^*$.
     - Exisitert nun $B \rightarrow w'$ mit $w' \in \Sigma^*$, füge B in Q und V' ein.
   - Wenn $Q = \emptyset$, dann weiter mit 2. Falls jedoch $S \notin V'$: alle Variablen nutzlos.

Wie erkennen wir nutzlose Variablen?

2. Berechnung von $V'' \subseteq V'$ aller $A \in V'$, für die es α, $\beta \in (V' \cup \Sigma)^*$ gibt, sodass eine Ableitung $S \rightarrow \alpha A \beta$ existiert, oder $A = S$.

Wie erkennen wir nutzlose Variablen?

2. Berechnung von $V'' \subseteq V'$ aller $A \in V'$, für die es α, $\beta \in (V' \cup \Sigma)^*$ gibt, sodass eine Ableitung $S \to \alpha A \beta$ existiert, oder $A = S$.
   - Füge zu allen Regeln $S \to \alpha A \beta$ mit α, $\beta \in (V' \cup \Sigma)^*$, $A \in V'$, A in V'' ein.

Wie erkennen wir nutzlose Variablen?

2. Berechnung von $V'' \subseteq V'$ aller $A \in V'$, für die es α, $\beta \in (V' \cup \Sigma)^*$ gibt, sodass eine Ableitung $S \rightarrow \alpha A \beta$ existiert, oder $A = S$.
   - Füge zu allen Regeln $S \rightarrow \alpha A \beta$ mit α, $\beta \in (V' \cup \Sigma)^*$, $A \in V'$, A in V'' ein.
   - Für jedes in V'' eingefügte A mit $A \rightarrow \alpha B \beta$, α, $\beta \in (V' \cup \Sigma)^*$ und $B \in V'$ füge auch B in V'' ein usw.

Wie erkennen wir nutzlose Variablen?

2. Berechnung von $V'' \subseteq V'$ aller $A \in V'$, für die es α, $\beta \in (V' \cup \Sigma)^*$ gibt, sodass eine Ableitung $S \rightarrow \alpha A \beta$ existiert, oder $A = S$.
    - Füge zu allen Regeln $S \rightarrow \alpha A \beta$ mit α, $\beta \in (V' \cup \Sigma)^*$, $A \in V'$, A in V'' ein.
    - Für jedes in V'' eingefügte A mit $A \rightarrow \alpha B \beta$, α, $\beta \in (V' \cup \Sigma)^*$ und $B \in V'$ füge auch B in V'' ein usw.
3. V'' sind die nützlichen Variablen

Satz: Endlichkeit kontextfreier Grammatiken

Für eine kontextfreie Grammatik G kann (in polynomialer Zeit) entschieden werden, ob $L(G)$ endlich ist.

Aufgabe 4

Ist die von der Grammatik $G = (V, \Sigma, S, R)$ mit $V = \{A, B, C, S\}$, $\Sigma = \{a, b\}$ und $R = \{$

$$\begin{array}{lcl} S & \rightarrow & AB|CA, \\ A & \rightarrow & a, \\ B & \rightarrow & BC|AB, \\ C & \rightarrow & aB|b \end{array}$$

$\}$
erzeugte Sprache endlich? Begründe Deine Antwort.

Wie kann gepfrüft werden, ob $G = (V, \Sigma, S, R)$ endlich ist?

- Nutzlose Variablen entfernen.

Wie kann gepfrüft werden, ob $G = (V, \Sigma, S, R)$ endlich ist?

- Nutzlose Variablen entfernen.
- Chomsky-Normalform erzeugen

Wie kann gepfrüft werden, ob $G = (V, \Sigma, S, R)$ endlich ist?

- Nutzlose Variablen entfernen.
- Chomsky-Normalform erzeugen
- Konstruiere Graph aus der Regelmenge
  - Knoten $:= V$
  - Kante zwischen Knoten A, B genau dann, wenn es eine Produktion gibt mit $A \to CB$ oder $A \to BC$, $A, B, C \in V$.

Wie kann gepfrüft werden, ob $G = (V, \Sigma, S, R)$ endlich ist?

- Nutzlose Variablen entfernen.
- Chomsky-Normalform erzeugen
- Konstruiere Graph aus der Regelmenge
  - Knoten $:= V$
  - Kante zwischen Knoten A, B genau dann, wenn es eine Produktion gibt mit $A \to CB$ oder $A \to BC$, $A, B, C \in V$.
- Suche im Graph nach Kreisen, falls keine vorhanden $\Rightarrow G$ ist endlich.

Warum muss man die nutzlosen Variablen entfernen, bevor man den Graph aufstellen kann?

Warum muss man die nutzlosen Variablen entfernen, bevor man den Graph aufstellen kann?

- es gibt keine Ableitungen, die die nutzlosen Variablen enthalten

Warum muss man die nutzlosen Variablen entfernen, bevor man den Graph aufstellen kann?

- es gibt keine Ableitungen, die die nutzlosen Variablen enthalten
- nutzlose Variablen können Kreise erzeugen (siehe obige Grammatik)

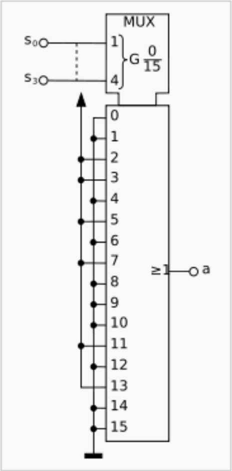
Überraschungsaufgabe

Gegeben ist eine Schaltfunktion $f(s3, s2, s1, s0)$, die genau dann 1 ist, wenn die Dualzahl $s3s2s1s0_2$ eine Primzahl ist.
Realisiere mit einem 3-MUX.
Das Problem ist dabei, dass die Funktion f vier Parameter hat, aber nur drei Steuersignale zur Verfügung stehen.

1. Wahrheitstafel von f aufstellen

Dez	s_3	s_2	s_1	s_0	a
0	0	0	0	0	0
1	0	0	0	1	0
2	0	0	1	0	1
3	0	0	1	1	1
4	0	1	0	0	0
5	0	1	0	1	1
6	0	1	1	0	0
7	0	1	1	1	1
8	1	0	0	0	0
9	1	0	0	1	0
10	1	0	1	0	0
11	1	0	1	1	1
12	1	1	0	0	0
13	1	1	0	1	1
14	1	1	1	0	0
15	1	1	1	1	0

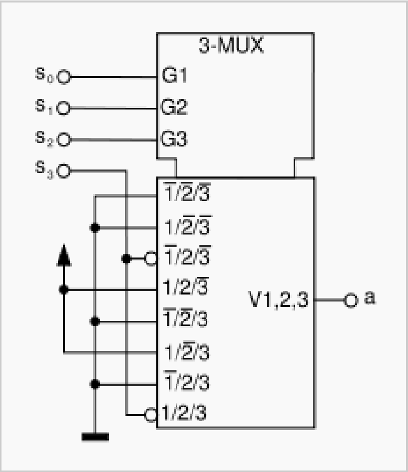
2. praktische Realisierung



3. neue Wahrheitstafel aufstellen

s_2	s_1	s_0	a
0	0	0	0
0	0	1	0
0	1	0	$\overline{S}_3$
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	$\overline{S}_3$

4. Schaltbild



Reflexion

Was haben wir heute gelernt?

Reflexion

Was haben wir heute gelernt?

- Pumping-/Odgen's-Lemma für kontextfreie Sprachen

Reflexion

Was haben wir heute gelernt?

- Pumping-/Odgen's-Lemma für kontextfreie Sprachen
- nutzlose Variablen entfernen

Reflexion

Was haben wir heute gelernt?

- Pumping-/Odgen's-Lemma für kontextfreie Sprachen
- nutzlose Variablen entfernen
- Entscheidbare Probleme für kontextfreie Sprachen

Noch Fragen?

Vorschau

Vorschau

- Kellerautomaten

Vorschau

- Kellerautomaten
- evt. Klausurwiederholung, daher bitte konkrete Wünsche per Email!

Bis zum nächsten Mal