7.1

$L = \{ w \in \{a,b\}^* \mid |w|_a = |w|_b \}$

$S \to aSbS \mid bSaS \mid \varepsilon$ ~~$(S \to SS)$~~

Beweis: Wir zeigen, für jedes $w \in L$ gilt $S \Rightarrow^* w$

Induktion über die Länge von w.

- Ind.-Anfang: $n = 0$, $w = \varepsilon$. $S \to \varepsilon$ ist Regel
  $\Rightarrow S \Rightarrow^* \varepsilon$
- Ind.-Schritt: $w \in L$ mit $|w| = 2n$ $(n \in \mathbb{N}_0)$

Fallunterscheidung:

- w fängt mit a an
  x sei das kürzeste Präfix von w, das auch in L ist

  $\underbrace{aabb}_{x}abbaabbaa = w \in L$

  $aabb = x$ kürzestes Präfix

  Es könnte $|w| = |x|$, $w = x$ sein $\Rightarrow w = a^n b^n$

  Dann: $S \Rightarrow aSbS \Rightarrow aaSbSbS$
  $\Rightarrow^* \underbrace{aaa\ldots a}_{n} S \underbrace{bSb\ldots b}_{n} S \Rightarrow^* \underbrace{a\ldots a}_{n} \underbrace{b\ldots b}_{n} = a^n b^n$

  Meistens: $|x| < |w| \Rightarrow w = xy$, y = Restwort

  $x = azb$ für ein Teilwort z, $w = azby$

  wobei $z, y \in L$

  $|z| < |w|$, $|y| < |w|$

  Nach Ind.-Voraussetzung können wir z, y ableiten

  $S \Rightarrow^* z$ $S \Rightarrow^* y$

  $\Rightarrow S \Rightarrow aSbS \Rightarrow^* azbS \Rightarrow^* azby = w$

- w fängt mit b an: analog.

7.2) $S \to ba \mid baS \mid BbbA$

$A \to a \mid aS \qquad B \to ab \mid b$

a) kontextfrei, nicht regulär (wegen $S \to \underline{B}bb\underline{A}$)

b) Umwandl. in Chomsky-Form

(1) ε-Regeln eliminieren (✓) $\quad A \to \varepsilon \quad S \to BbbA$

(2) Nukleose Symbole (✓) $\quad C \to CC, \; C \to Ca$

(3) Kettenregeln $\quad A \to B$

(4) Chomsky-NF

1. NT-Symbole C_a, C_b einführen

$\Rightarrow \quad S \to C_bC_a \mid C_bC_aS \mid BC_bC_bA$

$A \to a \mid C_aS \qquad B \to C_aC_b \mid b$

$C_a \to a \qquad C_b \to b$

2. Zwei Regeln nicht in Chomsky-NF

$\Rightarrow$ weitere NT-Symbole

$S \to C_bD_1, \quad D_1 \to CaS$

$S \to BD_2, \quad D_2 \to C_bD_3, \quad D_3 \to C_bA$

c) Sprache regulär?

Man kann B eliminieren: ~~$S \to BbbA, \; B \to ab \mid b$~~

$S \to abbbA \mid bbbA$

umgeformte Grammatik ist rechtslinear, also ist die Sprache regulär.

Fachhochschule Südwestfalen
University of Applied Sciences

Fachbereich: I+N

Prüfungsfach: GdI 3 (Probeklausur) | Prüfungsdatum: 17.12.2020

Name: | Matrikelnummer:

Note: | Unterschrift der/des Prüfenden:

Prof. Dr. Hans-Georg Eßer
GdI 3 (Probeklausur)

Klausur, 17.12.2020

1	2	3	4	5	6	7	8							Σ
5	6	8	5	8	5	4	5							46

(wird vom Prüfer ausgefüllt)

Hinweis: Bitte nutzen Sie die freien Bereiche auf den Aufgabenblättern (und bei Bedarf die leeren Rückseiten) für Ihre Lösungen. Punktezahl dieser Probeklausur: **46 P.** ($\approx$ 45 Minuten); echte Klausur: **90 P.**

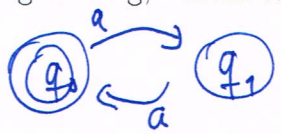
Es gilt wie üblich: $\mathbb{N} = \{1, 2, 3, \ldots\}$ und $\mathbb{N}_0 = \{0, 1, 2, 3, \ldots\}$ **Viel Erfolg!**

Aufgabe 1: Endliche Automaten (5 P.)

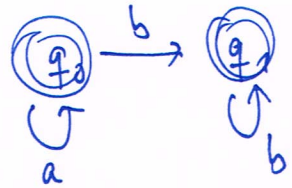
a) Geben Sie für die folgenden vier Sprachen *entweder* einen deterministischen endlichen Automaten an, der diese Sprache erkennt, *oder* eine Begründung, warum es keinen solchen Automaten gibt:

- $L_1 = \{a^{2n} \mid n \in \mathbb{N}_0\}$
- $L_2 = \{a^n b^m \mid n, m \in \mathbb{N}_0\}$
- $L_3 = \{a^n b^n \mid n \in \mathbb{N}_0\}$
- $L_4 = \{(ab)^n \mid n \in \mathbb{N}_0\}$

L_1:

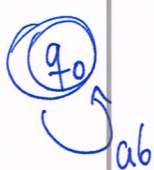


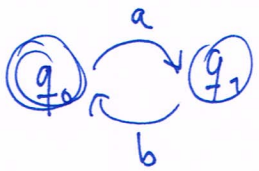
L_2:



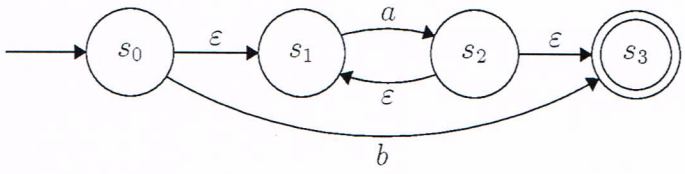
L_3 ↯ Klassiker der kontextfreien Sprachen:
Endliche Aut. können nicht zählen

L_4:





b) Welche Sprache erkennt der folgende endliche Automat mit ε-Transitionen?



$b \mid a^+$ $\qquad L = \{b\} \cup \{a^n \mid n \geq 1\}$

Aufgabe 2: Typ-3-Grammatiken (6 P.)

a) Die Definition **rechtslinearer Grammatiken** lässt Regeln der Form

$$A \to abcB$$

nicht zu; bei den so genannten **verallgemeinerten Typ-3-Grammatiken** sind diese erlaubt.

Erklären Sie, warum es keine Rolle spielt, ob wir rechtslineare Grammatiken (gemäß der ursprünglichen Definition) oder verallgemeinerte rechtslineare Grammatiken betrachten.

Verwenden Sie zur Argumentation das Beispiel von oben ($A \to abcB$) und wandeln Sie diese Regel in einen Satz von Regeln um, welche der ursprünglichen Definition rechtslinearer Grammatiken genügen.

$A \to abcB$ | $A \to aN_1$, $N_1 \to bN_2$, $N_2 \to cB$ | $A \Rightarrow aN_1 \Rightarrow abN_2 \Rightarrow abcB$

b) Geben Sie eine möglichst einfache (verallgemeinerte) rechtslineare Grammatik an, welche die Sprache $L = \{a^3b^nc^3 \mid n \in \mathbb{N}_0\}$ erkennt.

$S \to aaaB$
$B \to bB \mid ccc$

$S \to Bccc$
$B \to Bb \mid aaa$
linkslinear

Aufgabe 3: Kontextfreie Grammatiken / CYK (8 P.)

a) Die kontextfreie Grammatik G mit den Regeln

$$S \to aSb \mid ab$$

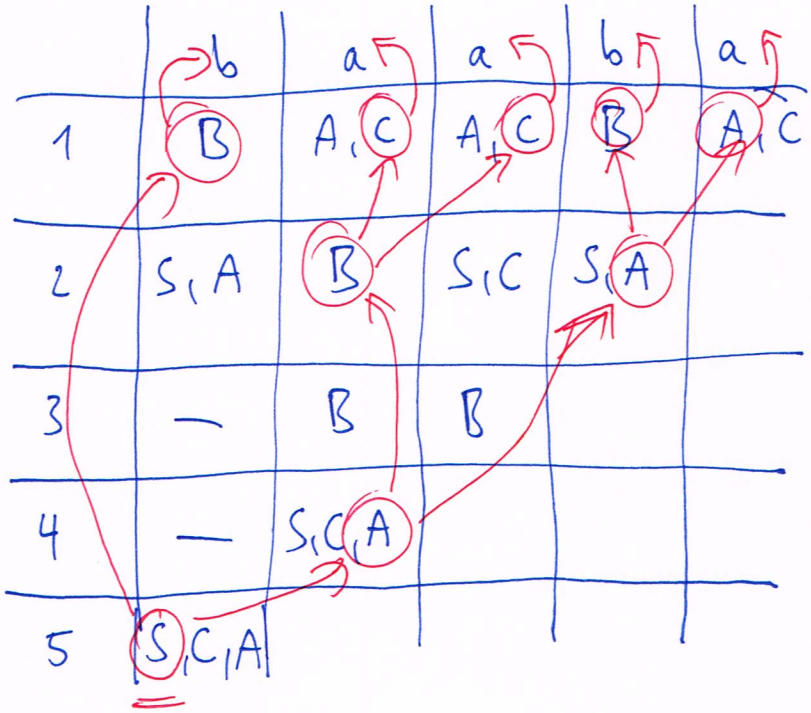
erzeugt bekanntlich die Sprache $L = \{a^n b^n \mid n \in \mathbb{N}\}$. Laut Vossen-Buch kann zu G ein äquivalenter Kellerautomat konstruiert werden, der diese Sprache erkennt. Welchen „Trick“ kann der Kellerautomat dabei nutzen, der einem endlichen Automaten ($\to$ reguläre Sprachen) versperrt ist?

Keller-A. kann zählen, schreibt für jedes gelesene "a" ein "a" auf den Keller und entfernt es beim Lesen von "b"

b) Verwenden Sie den CYK-Algorithmus, um für die folgende kontextfreie Grammatik in Chomsky-Normalform das Wortproblem für $w = baaba$ zu lösen:

$$S \to AB \mid BA,\ A \to BA \mid a,\ B \to CC \mid b,\ C \to AB \mid a$$

$S \to AB \mid BA$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$



$S \Rightarrow^* baaba$

Aufgabe 4: Kompression / Huffman (5 P.)

Erzeugen Sie anhand der durch die Zeichenkette

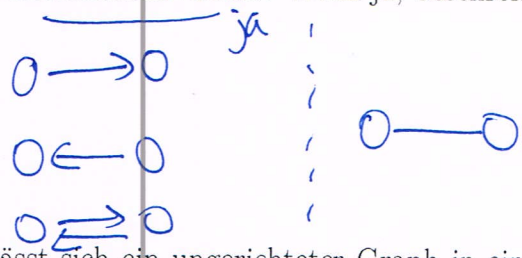
`Tri tra trulala`

gegebenen Häufigkeitsverteilung eine Huffman-Kodierung.

Aufgabe 5: Graphen: gerichtet vs. ungerichtet (8 P.)

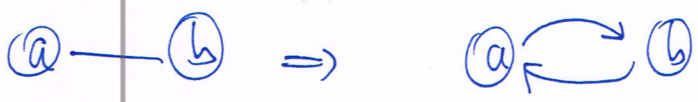
a) Wie lässt sich ein gerichteter Graph in einen ungerichteten Graphen umwandeln? Kommt es dabei zu Informationsverlust? Wenn ja, beschreiben Sie diesen.

ja



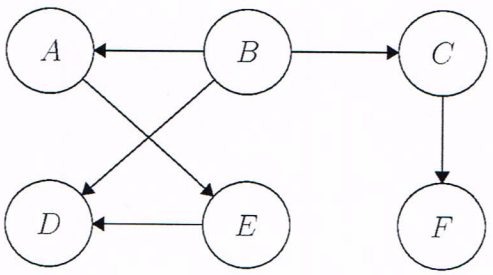
b) Wie lässt sich ein ungerichteter Graph in einen gerichteten Graphen umwandeln? Kommt es dabei zu Informationsverlust? Wenn ja, beschreiben Sie diesen.

Nein



c) Ein gerichteter Graph heißt **kreisfrei**, wenn es keinen gerichteten Kreis in diesem Graph gibt. Ein ungerichteter Graph heißt **kreisfrei**, wenn es keinen Kreis in diesem Graph gibt.

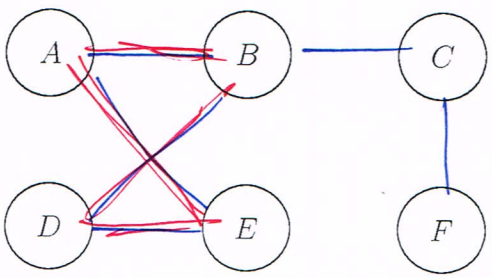
Betrachten Sie den folgenden gerichteten Graphen G:



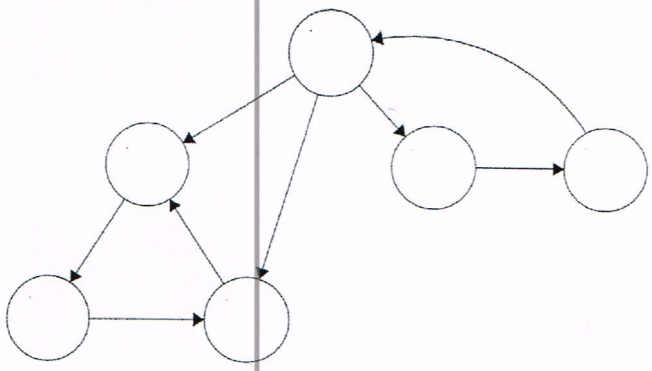
✓ kreisfrei

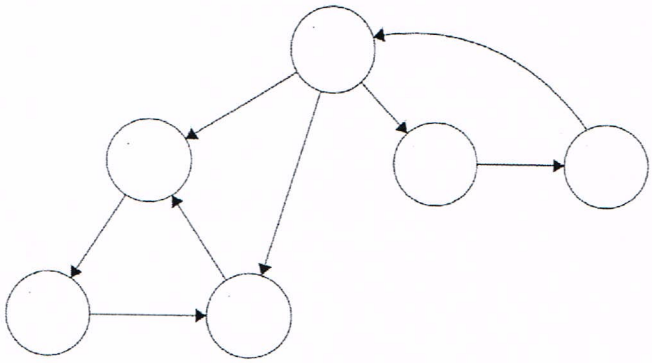
Stellen Sie fest, ob G kreisfrei ist (falls nein: Zeichnen Sie den Kreis nach).

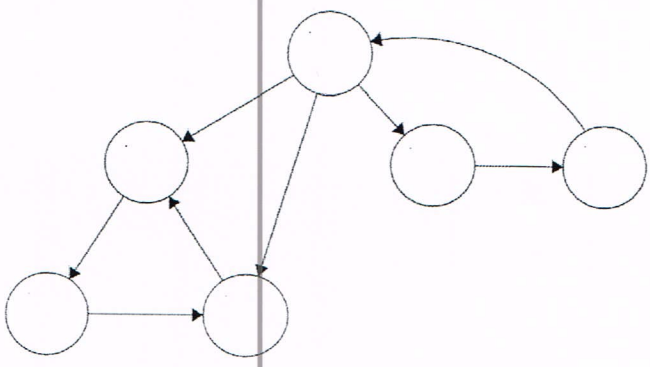
Wandeln Sie G in einen ungerichteten Graphen G' um (zeichnen!). Stellen Sie fest, ob G' kreisfrei ist (falls nein: Zeichnen Sie den Kreis nach).

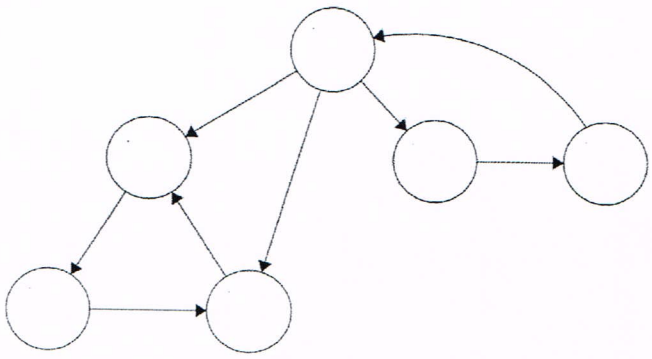


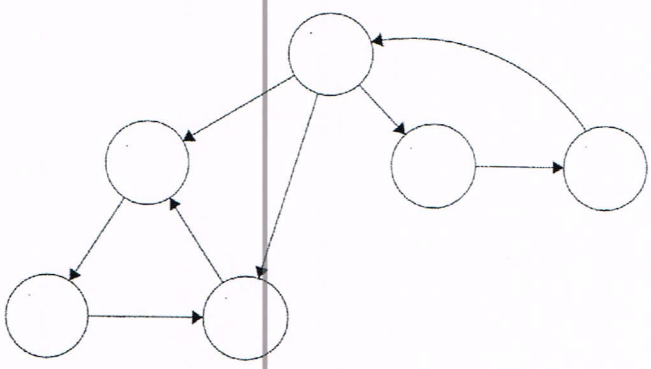
(Hilfsblatt zu Aufgabe 6)

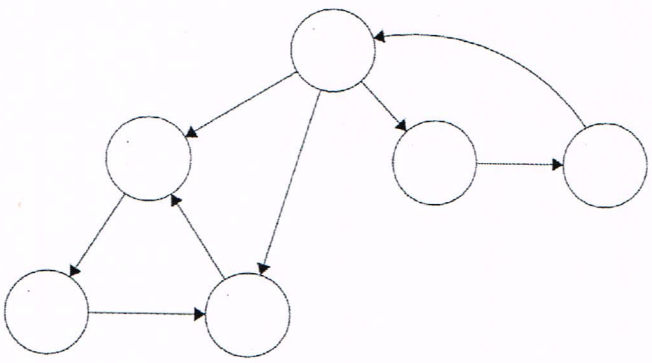


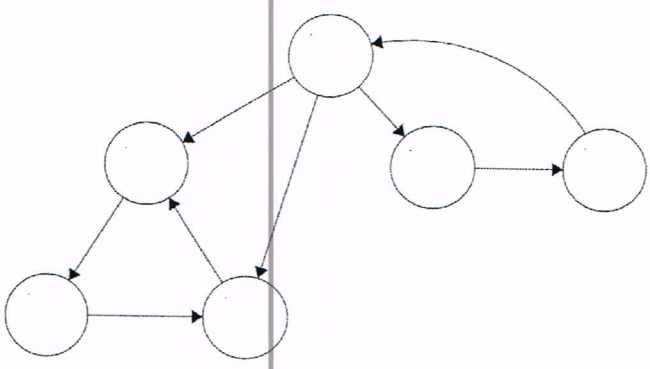


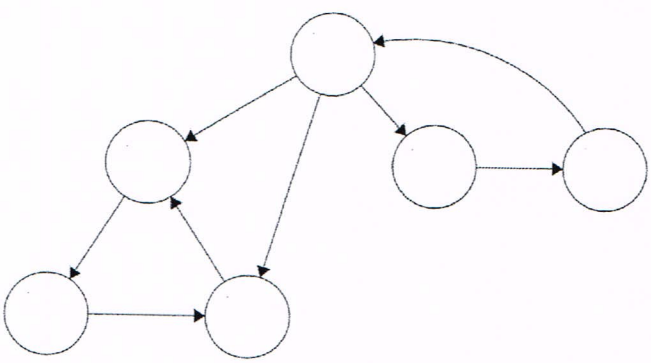






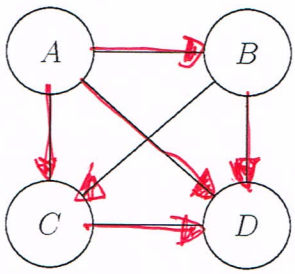






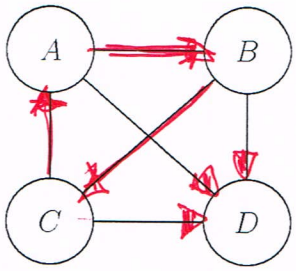
d) Betrachten Sie den vollständigen Graph G mit der Knotenmenge $\{A, B, C, D\}$. Wandeln Sie ihn (durch Einzeichnen von Pfeilen)

(i) in einen kreisfreien Graphen um:



Kreisfrei

(ii) in einen Graphen um, der genau einen Kreis enthält:

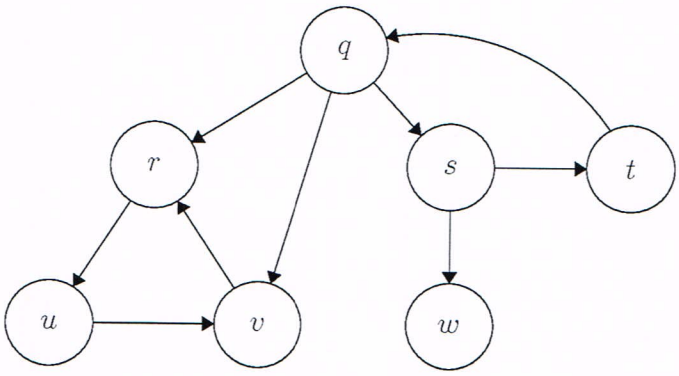


Genau 1 Kreis.

Aufgabe 6: Graphen: Tiefensuche (5 P.)

Führen Sie für folgenden Graphen G den Algorithmus DFS zur Tiefensuche durch; die Bearbeitung erfolgt in alphabetischer Reihenfolge der Knotenbezeichner (q, r, ...).

Klassifizieren Sie zudem alle Kanten (F = forward, B = backward, C = cross/ quer), die nicht Teil des durch DFS erzeugten Vorgängergraphen sind.



(Auf der folgenden Seite finden Sie einige Kopien des Graphs, die Sie zur Durchführung des Algorithmus verwenden *können* aber nicht müssen.)

Aufgabe 7: Rechnen in $\mathbb{Z}_n$ (4 P.)

a) Erstellen Sie für $\mathbb{Z}_{12}$ die Tabelle der multiplikativen Inversen. (Geben Sie also für jedes $z \in \mathbb{Z}_{12}$ entweder z^{-1} an oder tragen Sie einen Strich ein, wenn z^{-1} in $\mathbb{Z}_{12}$ nicht existiert.) Die allgemein gültige Lösung für $z = 0$ (es gibt kein 0^{-1}) ist bereits eingetragen.

z	0	1	2	3	4	5	6	7	8	9	10	11
z^{-1}	–	1	/	/	/	5	/	7	/	/	/	11

b) Nehmen Sie Stellung zur Frage: Gilt in $\mathbb{Z}_{12}$ die Aussage $3 < 4$ oder $4 < 3$ oder keine von beiden? Begründen Sie Ihre Antwort.

Aufgabe 8: Symmetrische vs. asymmetrische Verschlüsselung (5 P.)

a) Erläutern Sie den wesentlichen Unterschied zwischen symmetrischen und asymmetrischen Verschlüsselungsverfahren.

b) Erläutern Sie, wie das Kryptoverfahren **One-time Pad** funktioniert. Handelt es sich dabei um ein symmetrisches oder asymmetrisches Verfahren?