

Grundbegriffe der Informatik

Aufgabenblatt 6

Matr.nr.:

--	--	--	--	--	--	--

Nachname:

--

Vorname:

--

Tutorium: Nr.

--

Name des Tutors:

--

Ausgabe: 27. Januar 2017

Abgabe: 09. Februar 2017, 16:00h, 16:00 Uhr
im GBI-Briefkasten im Untergeschoss
von Gebäude 50.34

Lösungen werden nur korrigiert, wenn sie

- rechtzeitig,
- in Ihrer eigenen Handschrift,
- mit dieser Seite als Deckblatt und
- in der oberen **linken** Ecke zusammengeheftet

abgegeben werden.

Vom Tutor auszufüllen:

erreichte Punkte

Blatt 6:

	/ 35
--	------

(4 ECTS: 35)

Blätter 1 – 6:

	/ 203,5
--	---------

(4 ECTS: 166,5)

Ankündigung: Am 03.02. werden wir ein Extra-Übungsblatt herausgeben, mit dem man Bonuspunkte für den Übungsschein erreichen kann. D.h., die Anzahl der möglichen Punkte des Extra-Übungsblattes wird nicht für die zu erreichenden 50% der möglichen Gesamtpunkte der Übungsblätter hinzuge-rechnet werden. Das Blatt wird auch am 09.02. um 16:00h abgegeben werden müssen.

Aufgabe 6.1 (8 Punkte)

Der Merge-Sort Algorithmus ist ein Algorithmus zum Sortieren von Listen. Eine Implementierung des Algorithmuses ist unten in Pseudo-Code angegeben.

Geben Sie die Laufzeit des Algorithmus in Abhängigkeit von der Länge der Liste n möglichst genau in Θ -Notation an.

```
funktion mergesort(liste);
  if length(liste) <= 1
  then return liste
  else
    halbiere die liste in linkeListe, rechteListe
    linkeListe = mergesort(linkeListe)
    rechteListe = mergesort(rechteListe)
    return merge(linkeListe, rechteListe)

funktion merge(linkeListe, rechteListe);
  neueListe
  while (linkeListe und rechteListe nicht leer)
    if (linkeListe[0] <= rechteListe[0])
    then
      fuege linkeListe[0] in die neueListe hinten ein
      entferne es aus linkeListe
    else
      fuege rechteListe[0] in die neueListe hinten ein
      entferne es aus rechteListe

  while (linkeListe nicht leer)
    fuege linkeListe[0] in die neueListe hinten ein
    entferne es aus linkeListe
  while (rechteListe nicht leer)
    fuege rechteListe[0] in die neueListe hinten ein
    entferne es aus rechteListe
  return neueListe
```

Aufgabe 6.2 (2+2+2+2+2=10 Punkte)

Für jeden gerichteten Graphen $G = (V, E)$ ist der sogenannte *Kantengraph* (engl. *line graph*) $L(G) = (V', E')$ wie folgt definiert: Wenn E nicht leer ist, dann ist

$$V' = E,$$

$$E' = \{((x, y), (y, z)) \mid (x, y), (y, z) \in V'\};$$

wenn E leer ist, dann ist $V' = \{0\}$ und $E' = \{\}$.

Für $n \in \mathbb{N}_0$ sei der n -te iterierte Kantengraph $L^n(G)$ so definiert:

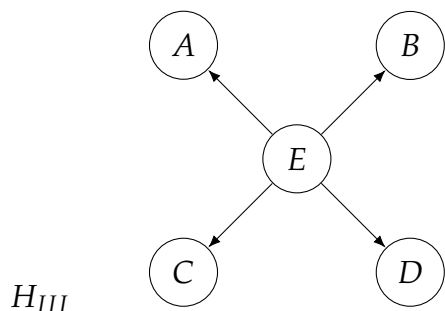
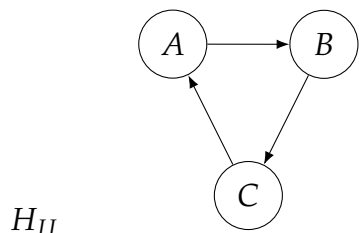
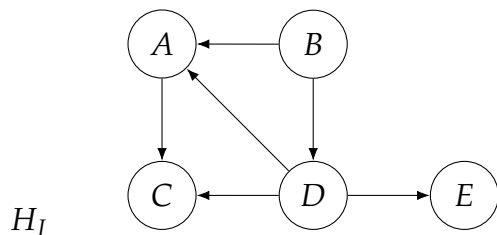
$$L^0(G) = G,$$

$$\forall n \in \mathbb{N}_0 : L^{n+1}(G) = L(L^n(G)).$$

Es bezeichne im folgenden $L_V^n(G)$ die Knotenmenge von $L^n(G)$ und $L_E^n(G)$ die Kantenmenge von $L^n(G)$.

Hinweis: $|M|$ bezeichnet im folgenden stets die Kardinalität, also die Anzahl der Elemente, einer endlichen Menge M .

a) Zeichnen Sie zu den folgenden Graphen H_I , H_{II} und H_{III}



die Kantengraphen $L(H_x)$ und benennen sie jeweils die Knoten sinnvoll.

b) Zeichnen Sie ausgehend von der Adjazenzmatrix A den Graphen H_A , sowie den Kantengraphen $L(H_A)$:

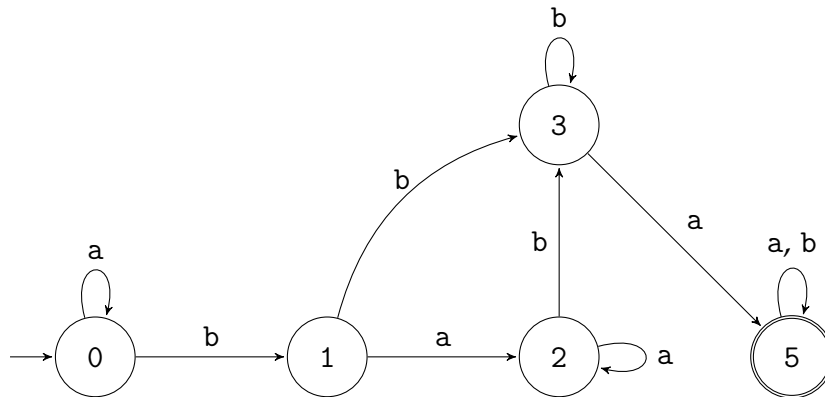
$$A = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \end{pmatrix}$$

- c) Zeichnen Sie ausgehend von der Adjazenzmatrix B den iterierten Kantengraphen $L^2(H_B)$:

$$B = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

Aufgabe 6.3 (2 + 1,5 + 1,5 = 5 Punkte)

Der endliche Akzeptor $A = (Z, z_0, X, f, F)$ sei gegeben durch



- Geben Sie die von A akzeptierte Sprache $L(A)$ unter ausschließlicher Benutzung der formalen Sprachen $\{a\}$, $\{b\}$ sowie $\{a, b\}$, des Konkatenationsabschlusses, des ϵ -freien Konkatenationsabschlusses und des Produkts formaler Sprachen an.
Beispiel: $\{a, b\}^* \cdot \{a\} \cdot \{b\}$
- Geben Sie graphisch einen endlichen Akzeptor B mit vier Zuständen an, der dieselbe formale Sprache wie A akzeptiert.
- Geben Sie graphisch einen endlichen Akzeptor C mit vier Zuständen an, von denen zwei akzeptierend sind, der die formale Sprache $L(C) = \{a\}^* \cdot \{b\} \cdot \{a\}^* \cdot \{b\}^+ \cdot \{a, b\}^*$ akzeptiert.

Aufgabe 6.4 (2+3+3+4=12 Punkte)

Konstruieren Sie jede der folgenden formalen Sprachen $L_i \subseteq \{a, b\}^*$ jeweils einen regulären Ausdruck R_i mit $\langle R_i \rangle = L_i$.

- $L_1 = \{w \in \{a, b\}^* \mid w \text{ enthält das Teilwort } abb\}$
- $L_2 = \{w \in \{a, b\}^* \mid w \notin L_1\}$
- $L_3 = \{w \in \{a, b\}^* \mid w \text{ enthält genau zweimal das Teilwort } abb\}$
- $L_4 = \{w \in \{a, b\}^* \mid w \notin L_3\}$