Grundbegriffe der Informatik
Aufgabenblatt 7

Matr.nr.:							

Nachname:

Vorname:

Tutorium: Nr.

Name des Tutors:

Ausgabe: 4. Februar 2017

Abgabe: 9. Februar 2017, 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

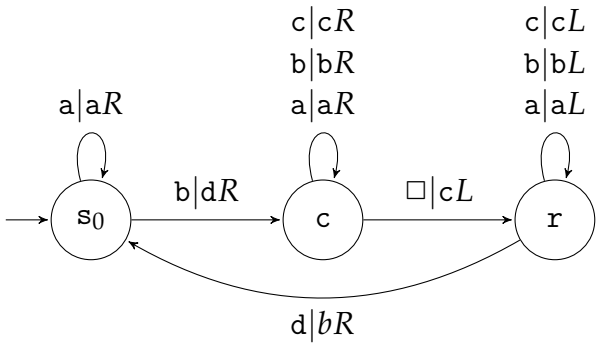
abgegeben werden.

Vom Tutor auszufüllen:

erreichte Punkte

Blatt 7:	/ 23	(4 ECTS: 34)
Blätter 1 – 7:	/ 23	(4 ECTS: 0)

Aufgabe 7.1 (2 + 2 + 2 + 2 = 8 Punkte)



Gegeben sie die oben gezeichnete Turing-Maschine T. Die Eingabe sein ein beliebiges Wort $w \in \{\mathtt{a}, \mathtt{b}\}^+$.

a) Für welche Eingabewörter hält T an?
b) In welchen Zuständen kann T für eine Eingabe anhalten?
c) Welches Wort steht für die Eingabe w auf dem Band?
d) Geben Sie eine Funktion f an mit $\text{Time}_T \in \Theta(f)$.

Lösung 7.1

a) Für alle.
b) In $\mathtt{s}_0$.
c) Das Wort wv mit $v = c^k$, wobei $k =$ Anzahl von b in w.
d) Im schlimmsten Fall muss ein Wort, dass nur aus b besteht abgeeignet werden. Dann gilt $\text{Time}_T \in \Theta(n^2)$

Aufgabe 7.2 (1 + 2 + 2 + 3 = 8 Punkte)

Ein weiterer Sotieralgorithmus ist der Bucket-Sort-Algorithmus. Der Pseudocode des Algorithmuses lautet:

```
bucket_sort(l)
  min = minElement(l)
  max = maxElement(l)
  buckets = array(max–min)
  foreach (e in l)
       buckets[ e] = true
  r = []
  for i in (min,max)
       if(buckets[i])
            r.append(i)
  return r
```

- Geben Sie die untere und obere Schranke für die Laufzeit von minElement und maxElement an, falls es optimal implementiert ist.
- Geben Sie die untere und obere Schranke des Bucket-Sort-Algorithmus an.
- Unter welchen Bedinungen ist der Bucketsortalgorithmus schneller als der Merge-Sort-Algorithmus?
- Nennen Sie 3 Nachteile dieser Implementierung des Bucket-Sort-Algorithmus.

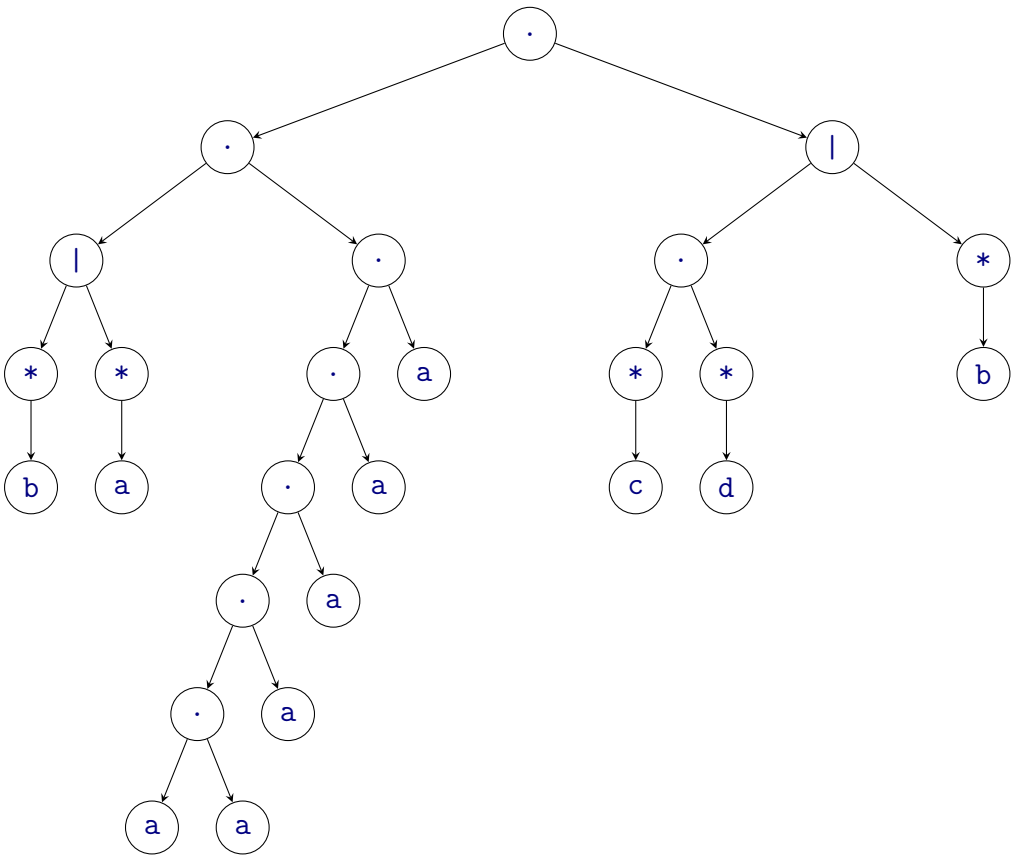
Lösung 7.2

- Geben Sie die untere und obere Schranke für die Laufzeit von minElement und maxElement an, falls es optimal implementiert ist?
  - Jedes element einmal anschauen -> $\Theta(len(l))$
- Geben Sie die untere und obere Schranke des Bucket-Sort algorithmuses an
  - Jedes Element einmal anschauen und jedes Bucket einmal anschauen -> $\Theta(len(l) + (max - min))$
- Unter welchen Bedinungen ist der Bucketsortalgorithmus schneller als der Merge-Sort-Algorithmus
  - max-min < len(l)*log(len(l))
- Nennen Sie 3 Nachteile dieser Implementierung des Bucket-Sort-Algorithmus
  - Hoher Speichverbrauch
  - Nur für natuerliche Zahlen
  - Nur für Listen ohne doppelte Werte

Aufgabe 7.3 (1 + 2 = 3 Punkte)

- Geben Sie den Regex-Baum für den regulären Ausdruck $R :=$ `(b*|a*)(aaaaaa)(c*d*|b*)` an.
- Geben Sie für den regulären Ausdruck R aus Teilaufgabe a) einen anderen regulären Ausdruck an, der auch $\langle R \rangle$ erzeugt und einen Regex-Baum benötigt, der niedriger ist als der Regex-Baum aus Teilaufgabe a).

Lösung 7.3

- 

- $R_2 =$ `(b*|a*)((aaa)(aaa))(c*d*|b*)`

Aufgabe 7.4 (4 Punkte)

Ein ganzes Wintersemester haben die Studierenden der Informatik am KIT in Sorge darauf gewartet. Sie dachten schon, dass sie diesmal von diesem Schicksal veschont würden. Aber, zu früh gefreut! Der Schrecken findet eine Forsetzung: **Dr. Meta ist wieder da!**

Viele hier fragen sich sofort: "Dr. wer?", während anderen eher "Spinnen die hier jetzt völlig?" durch den Kopf geht. Aber seit gewarnt ihr Unwissenden! Die Chroniken der Grunbegriffe der Informatik sind voll mit den Schandtaten dieses wissenschaftlichsten aller Erzbösewichte.

Um seiner Rückkehr die notwenige Aufmerksamkeit zu verschaffen, hat Dr. Meta, mit Hilfe seines Gehilfen Nr. 20, dreizehn Studierende aus der Vorlesung Grundbegriffe der Informatik in seine Gewalt gebracht. Eigentlich war ihr Schicksal schon besiegelt. Da aber Dr. Meta während seiner Auszeit zu viele James-Bond-Filme geschaut hat, gibt er den Studierenden eine letze, in seinen Augen aussichtslose Chance. Er stellt sie vor folgendes Problem, wofür sie, wenn sie es lösen, sogar eine große Kiste voll Gold bekommen. (Ein Tresorraum voller Gold ala Fort Knox wie in Goldfinger steht Dr. Meta leider nicht zur Verfügung. So erfolgreich als Superschurke war er bisher dann doch nicht.)

Jeder Studierende wird in eine separate Zelle gesperrt und hat niemals Kontakt zu seinen Kommilitonen. Es wird zufällig ein Studierender ausgewählt, wel-

cher in den Raum mit der Goldkiste geführt wird. Nun hat er die Wahl, ob er sie öffnen, schließen oder so belassen will, wie er sie vorfand. Danach wird er wieder in seine Zelle geführt und es wird wieder ein zufälliger Studierender ausgelost.

Dieser Prozess wiederholt sich, bis irgend ein Studierender sich meldet und behauptet, dass nun jeder Studierende mindestens einmal im Raum mit der Goldkiste war. Stimmt seine Behauptung so sind sie frei und dürfen mit dem Schatz von dannen ziehen.

Bevor die Studierende in die Einzelzellen gesperrt werden, dürfen sie einen Plan aushecken und ihnen ist bekannt, ob die Kiste zu Beginn geschlossen oder offen ist.

Zu Dr. Metas Entsetzen lösen die Studierende die Aufgabe. Aber es kommt eine neues Wintersemester...

Mit welche Strategie haben die angehenden Informatiker die Aufgabe gelöst?

Lösung 7.4

Annahme die Kiste ist zu Beginn geschlossen.

Es wird ein Zählstudierender ausgemacht. Wenn er den Raum betritt und eine geöffnete Kiste vorfindet, schließt er sie und zählt eins weiter, ist sie geschlossen, lässt er sie so.

Kommt einer der anderer Studierender in die Kammer, so öffnet er die Kiste nur wenn sie zu ist und er sie noch nie geöffnet hat. Ansonsten lässt er sie immer in dem Zustand in dem sie sich momentan befindet.

Hat der Zählstudierende die Truhe 12 mal geschlossen so kann er sich sicher sein, das jeder seiner Kommilitonen mindestens einmal im Raum war und Beendet den Prozess.