Übungsblatt 5
DEAs und NEAs

Theoretische Informatik
Studiengang Angewandte Informatik
Wintersemester 2015/2016
Prof. Barbara Staehle, HTWG Konstanz

Aufgabe 5.1

[**topic = Erweiterungen von** A_1]

> *Allgemein bezüglich Überführung von Automaten!*
> *DEA, jeder Knoten hat* ***höchstens*** *einen Übergang (oder keinen) mit der eine Beschriftung* (q, a, q')*,anderenfalls spricht man von NEA....*

Wie betrachten im folgenden die Automaten A_2, A_3, A_4, welche Abänderungen des aus der Vorlesung bekannten Automaten A_1 sind. In Abbildung /refA234 ist jeweils ihr Zustandsübergangsdiagramm dargestellt.

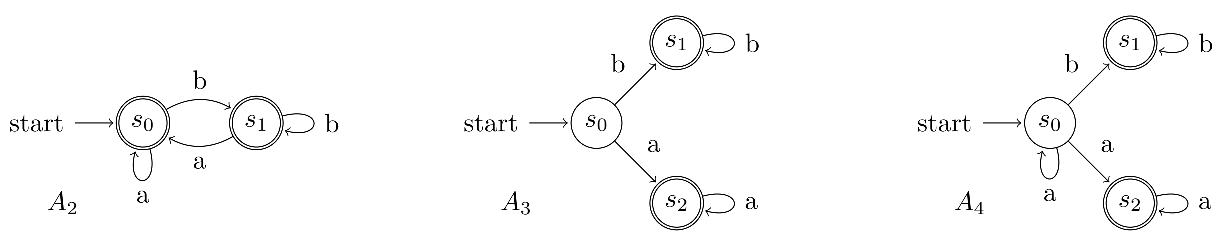


Abbildung 1: Zustandsübergangsdiagramme von A_2, A_3, A_4

Teilaufgabe 5.1.1

[credits = 1]

Geben Sie für jeden der Automaten an, ob er ein deterministischer endliche Akzeptor, oder ein nichtdeterministischer endlicher Akzeptor ist.

Lösung :

- A_2 ist DEA
- A_3 ist DEA
- A_4 ist NEA, von S0 gehen zwei Übergänge mit einer Beschriftung nämlich a

Teilaufgabe 5.1.2

[credits = 2]

Geben Sie für jeden der Automaten

1. die Zustandsmenge,
2. die Finalmenge,
3. die Zustandsübergangsfunktion in tabellarischer Form an. (für NEAs,alle in Mengen Klammern wenn es keinen Übergang gibt dann schreibt man $\varnothing$, bei DEAs schreibt man -)

Lösung :

Allgemein Zustandmenge : $A = \{S, \Sigma, \sigma, F, start\}$

- $A_2 = (\{s_0, s_1\}, \{a, b\}, \sigma, \{s_0, s_1\}, s_0)$
  mit $\sigma =$

S/Σ	a	b
s_0	s_0	s_1
s_1	s_0	s_1

- $A_3 = (\{s_0, s_1, s_2\}, \{a, b\}, \sigma, \{s_1, s_2\}, s_0)$
  mit $\sigma =$

S/Σ	a	b
s_0	s_2	s_1
s_1	$-$	s_1
s_2	s_2	$-$

- $A_4 = (\{s_0, s_1, s_2\}, \{a, b\}, \sigma, \{s_1, s_2\}, s_0)$
  mit $\sigma =$

S/Σ	a	b
s_0	$\{s_0, s_2\}$	$\{s_1\}$
s_1	$\varnothing$	$\{s_1\}$
s_2	$\{s_2\}$	$\varnothing$

Teilaufgabe 5.1.3

[credits = 2,]

Geben Sie für jeden der Automaten (wenn möglich) 3 Worte über dem Alphabet $\Sigma = \{a, b\}$ an Benutzen Sie Ihre Ergebnisse, um für jeden Automaten dessen akzeptierte Sprache anzugeben.

1. die akzeptiert werden,
   Lösung :

   - Für Automat A_2
     $w_2 \in \mathcal{L}_2$
     $w_2 = aaabbbb$ mit $\mathcal{L}_2 = \{a^n b^m | n, m \in \mathbb{N}\}$
     $w_2 = aaababb$ mit $\mathcal{L}_2 = \{a^n b^m a^n b^m | n, m \in \mathbb{N}\}$
     $w_2 = bb$ mit $\mathcal{L}_2 = \{a^n b^m | n \in \mathbb{N}_0, m \in \mathbb{N}\}$
   - Für Automat A_3
     $w_3 \in \mathcal{L}_3$
     $w_3 = aaa$ mit $\mathcal{L}_3 = \{a^n | n \in \mathbb{N}\}$
     $w_3 = bbbb$ mit $\mathcal{L}_3 = \{b^m | m \in \mathbb{N}\}$
     $w_3 = a$ mit $\mathcal{L}_3 = \{a^n | n = 1\}$
   - Für Automat A_4
     $w_4 \in \mathcal{L}_4$
     $w_4 = aaa$ mit $\mathcal{L}_4 = \{a^n | n \in \mathbb{N}\}$
     $w_4 = bbbb$ mit $\mathcal{L}_4 = \{b^m | m \in \mathbb{N}\}$
     $w_4 = abb$ mit $\mathcal{L}_4 = \{a^n b^m | n, m \in \mathbb{N}\}$

2. die nicht akzeptiert werden.
   Lösung :

- Für Automat A_2
  $w_2 \notin \mathcal{L}_2$
  Keine Wörter !
- Für Automat A_3
  $w_3 \notin \mathcal{L}_3$
  $w_3 = ba$
  $w_3 = ab$
  $w_3 = aab$
- Für Automat A_4
  $w_4 \notin \mathcal{L}_4$
  $w_4 = aaba$
  $w_4 = ba$
  $w_4 = aba$

Teilaufgabe 5.1.4

[credits = 2,]
Geben Sie für jeden Automaten $\hat{\delta}(s_0, bbb)$ (also den Zustand, in welchem sich der Automat nach Einlesen des Wortes *bbb* befindet) an.
Lösung:

- Für Automat A_2
  $\sigma(s_0, bbb) \to \sigma(s_1, bb) \to \sigma(s_1, b) \Rightarrow s_1$
- Für Automat A_3
  im zustand s_1
- Für Automat A_4
  im Zustand s_1

Aufgabe 5.2

[**[topic = Paritätscode**]
Zur Fehlererkennenung bei einer Datenübertragung wird oft der *Paritätscode* genutzt.

Die Grundidee des Paritätscodes ist, ausschließlich Datenpakete zu versenden, die eine gerade Anzahl Einsen aufweisen. Hierzu werden die Datenpakete vor dem Versenden um ein Paritätsbit ergänzt, das die Gesamtzahl der Einsen bei Bedarf gerade werden lässt.

Ein Übertragungsfehler wird dadurch erkannt, dass die Anzahl der Einsen ungerade ist. Ist die Anzahl der Einsen gerade, wurde das Pakte korrekt übertragen.

Das vor der Übertragung hinzuzufügende Paritätsbit berechnet sich wie folgt:

- 1, falls die Anzahl der Einsen im Datenpaket ungerade ist
- 0, falls die Anzahl der Einsen im Datenpaket gerade ist

Ein Paritätscode der Länge 4 versieht also die ersten 3 Datenbits mit einem 4. Paritätsbit, so dass die Pakete insgesamt wie folgt aussehen:

- 000|0
- 001|1
- 010|1
- ...

Teiaufgabe 5.2.1

[credits = 1]

Komplettieren Sie die Liste, bis Sie alle 8 verschiedenen Codewörter der Länge 3 um ihr Paritätsbit ergänzt haben.

Lösung

- 000|0
- 001|1
- 010|1
- 011|0

- 100|1
- 101|0
- 110|0
- 111|1

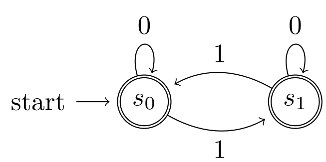
Teiaufgabe 5.2.2

[credits = 2]

Konstruieren Sie einen DEA A_P, der die Integrität eines empfangenen Datenpakets überprüft und alle korrekt übertragenen Wörter akzeptiert. Wurde ein einzelnes Bit des Datenpakets während der Übertragung verfälscht, so soll der Automat das Eingabewort ablehnen.

Lösung

$A_p = (\{s_0, s_1\}, \{0, 1\}, \sigma, \{s_1\}, s_0)$
mit $\sigma =$



A_p

Teiaufgabe 5.2.3

[topic = , credits = 1]

Wie verhält sich der von Ihnen konstruierte Automat, falls zwei Bits während der Datenübertragung verfälscht wurden? Weist er dieses falsche Wort auch zurück? Begründen Sie Ihre Aussage.

Lösung

Wenn richtig verstanden hab, also wenn 2 Bits verfälscht werden, könnte eine akzeptierte Bitfolge entstehen, wenn es der Fall ist, dann akzeptiert der Automat die entstandene Bitfolge, sollte die Verfälschung eine nicht akzeptierte

Bitfolge entstehen lassen, dann wird der Automat diese ablehnen !

Beispiel : 0000, Verfälschung von 2 Bits → 0101, was es zu Akzeptanz folgt.

Aufgabe 5.3

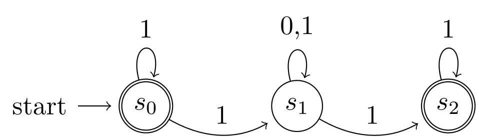
[**topic = Die Sprache** L_{11}]
Gegeben sei die Sprache L_{11}, die alle Wörter über $\{0, 1\}$ enthält, die mit einer 1 beginnen und mit einer 1 enden. Formal:

$$L_{11} = \{\omega \in \{0,1\}^* | \exists\ u \in \{0,1\}^* : \omega = 1u1\}$$

Teiaufgabe 5.3.1

[credits = 2] Geben Sie einen NEA N_{11} an der L_{11} akzeptiert, für den also $\mathcal{L}(N_{11}) = L_{11}$
Lösung :Sternchen heißt beliebig viel oder leer!!!!

$N_{11} = (\{s_0, s_1, s_2\}, \{0, 1\}, \sigma, \{s_0, s_2\}, s_0)$
mit $\sigma =$

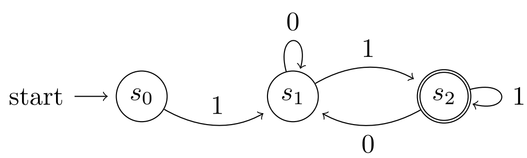


N_{11}

Teiaufgabe 5.3.2

[credits = 2]
Geben Sie einen DEA A_{11} an, welcher L_{11} akzeptiert. Tipp: Dieser benötigt genauso viele Zustände wie N_{11}, jedoch mehr Zustandsübergänge.
Lösung :

$N_{11} = (\{s_0, s_1, s_2\}, \{0, 1\}, \sigma, \{s_2\}, s_0)$
mit $\sigma =$



N_{11}

Teiaufgabe 5.3.3

[credits = 2]

Geben Sie einen regulären Ausdruck an, der L_{11} erzeugt. Hierbei ist es egal, ob Sie eine formale, oder eine Unix-ähnliche Syntax wählen.

Lösung

$1(01)^*1^*$

Teilaufgabe 5.3.4

[credits = 2]

Geben Sie eine Grammatik G_{11} an, die L_{11} erzeugt.

$G_N = \{\{S, A\}, \{0, 1\}, P, S\}$ mit P =
$S \rightarrow 1A1$
$A \rightarrow 0A|1A|\epsilon$