LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Formale Sprachen und Komplexität
Theoretische Informatik für Studierende der Medieninformatik
Sommersemester 2024

Zentralübung 4

Prof. Dr. Jasmin Blanchette

Lehr- und Forschungseinheit für
Theoretische Informatik und Theorembeweisen

Stand: 19. Juni 2024
Basierend auf Folien von PD Dr. David Sabel

Plan für heute

1. Minimierung von DFAs
2. Kellerautomaten (PDAs)
3. Turingmaschinen

1. Minimierung von DFAs

Minimierung eines DFAs mit dem tabellarischen Ansatz

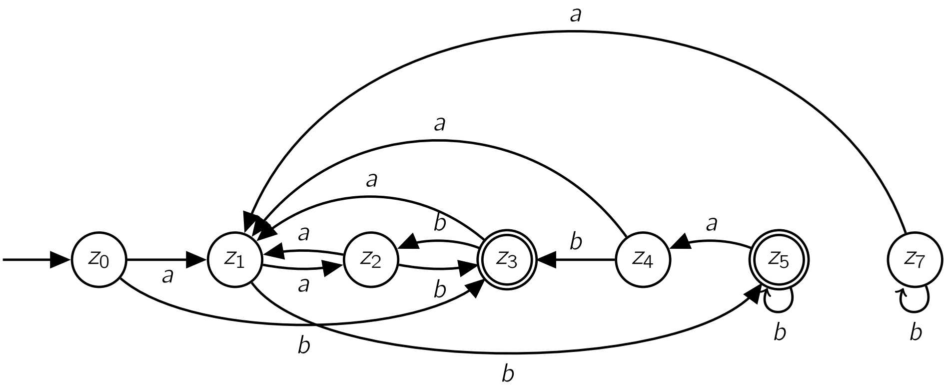
Sei M ein DFA.

Intuitiver Ansatz:

1. Entferne alle nicht erreichbaren Zustände von M.
2. Konstruiere die Partitionstabelle.
3. Bilde den minimalen DFA M', indem Zustände derselben Klasse verschmolzen werden, basiert auf der letzten Reihe der Partitionstabelle.

Aufgabe: DFA minimieren

Minimieren Sie den folgenden DFA:



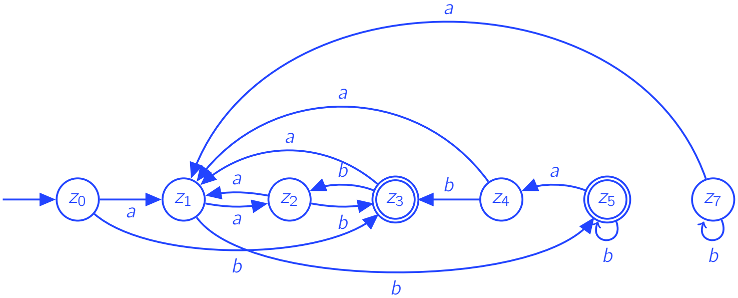
Aufgabe: DFA minimieren

Antwort:

Aufgabe: DFA minimieren

Antwort:

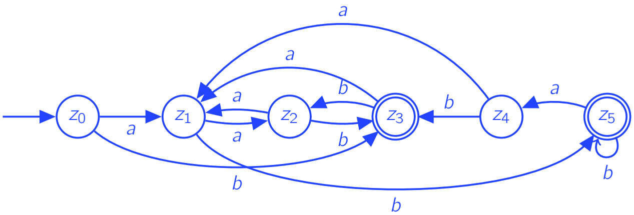
1. Entferne alle nicht erreichbaren Zustände.



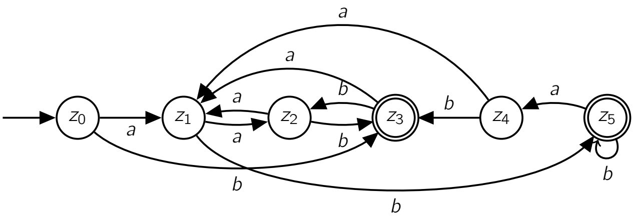
Aufgabe: DFA minimieren

Antwort:

1. Entferne alle nicht erreichbaren Zustände.

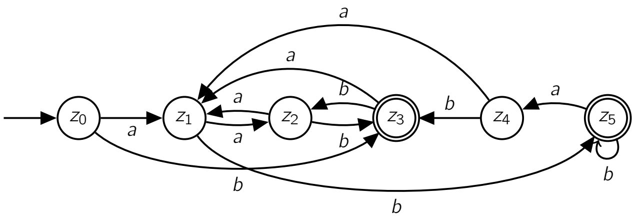


Aufgabe: DFA minimieren



2. Konstruiere die Partitionstabelle.

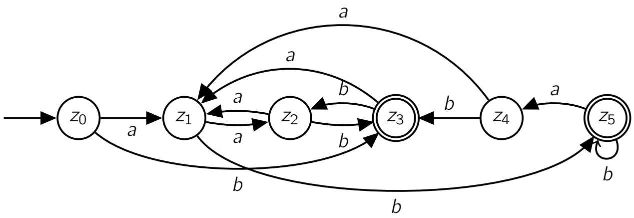
Aufgabe: DFA minimieren



2. Konstruiere die Partitionstabelle.

0. | z_0 z_1 z_2 z_4 | z_3 z_5 |

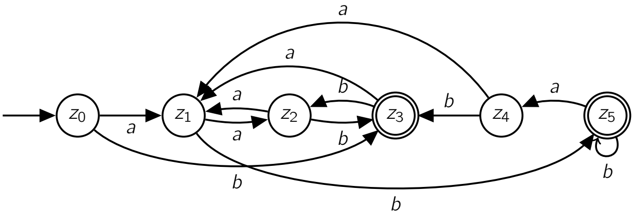
Aufgabe: DFA minimieren



2. Konstruiere die Partitionstabelle.

0.	z_0	z_1	z_2	z_4 |	z_3	z_5	
1.	z_0	z_1	z_2	z_4 |	z_3 |	z_5	mit b

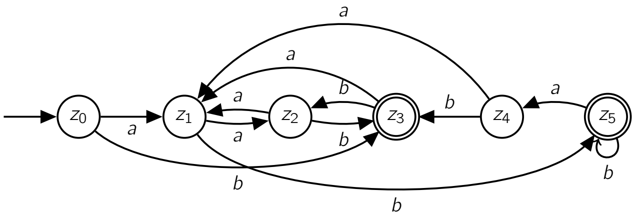
Aufgabe: DFA minimieren



2. Konstruiere die Partitionstabelle.

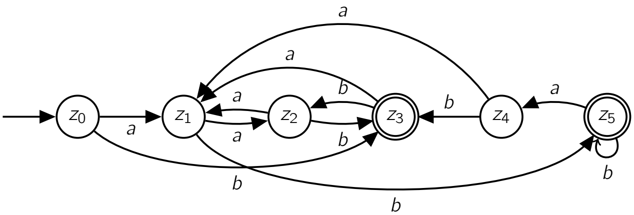
0.	z_0	z_1	z_2	z_4 |	z_3	z_5	
1.	z_0	z_1	z_2	z_4 |	z_3 |	z_5	mit b
2.	z_0	z_2	z_4 |	z_1 |	z_3 |	z_5	mit b

Aufgabe: DFA minimieren



3. Bilde den minimalen DFA, indem Zustände derselben Klasse verschmolzen werden, basiert auf der letzten Reihe der Partitionstabelle.

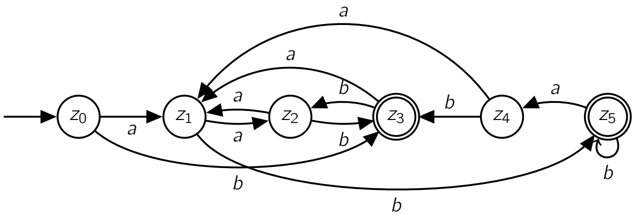
Aufgabe: DFA minimieren



3. Bilde den minimalen DFA, indem Zustände derselben Klasse verschmolzen werden, basiert auf der letzten Reihe der Partitionstabelle.

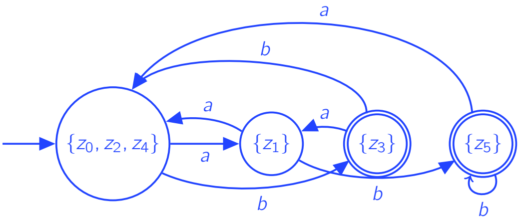
z_0 z_2 z_4	z_1	z_3	z_5

Aufgabe: DFA minimieren



3. Bilde den minimalen DFA, indem Zustände derselben Klasse verschmolzen werden, basiert auf der letzten Reihe der Partitionstabelle.

z_0 z_2 z_4			z_1	z_3	z_5



2. Kellerautomaten (PDAs)

Definition eines Kellerautomaten

Definition

Ein (nichtdeterministischer) Kellerautomat (pushdown automaton, PDA) ist ein 6-Tupel $M = (Z, \Sigma, \Gamma, \delta, z_0, \#)$, wobei:

- Z ist eine endliche Menge von Zuständen
- Σ ist das (endliche) Eingabealphabet
- Γ ist das (endliche) Kelleralphabet
- $\delta : Z \times (\Sigma \cup \{\varepsilon\}) \times \Gamma \to \mathcal{P}_e(Z \times \Gamma^*)$ ist die Überführungsfunktion
- $z_0 \in Z$ ist der Startzustand
- $\# \in \Gamma$ ist das Startsymbol im Keller.

1. Quiz

Eine Konfiguration eines Kellerautomaten ist ein Tupel (z, w, W).
Was sind z, w und W?

1. Quiz

Eine Konfiguration eines Kellerautomaten ist ein Tupel (z, w, W).
Was sind z, w und W?

Antwort:

- z ist der aktuelle Zustand
- w ist die Resteingabe
- W ist der aktuelle Kellerinhalt.

2. Quiz

Sei $(z', B_1 \cdots B_k) \in \delta(z, a, A)$. Dann gilt $(z, aw, AW) \vdash$???.
Was ist für ??? einzusetzen?

2. Quiz

Sei $(z', B_1 \cdots B_k) \in \delta(z, a, A)$. Dann gilt $(z, aw, AW) \vdash ???$.
Was ist für ??? einzusetzen?

Antwort: $(z', w, B_1 \cdots B_k W)$.

3. Quiz

Sei $(z', B_1 \cdots B_k) \in \delta(z, \varepsilon, A)$. Dann gilt $???_1 \vdash ???_2$.
Was ist für $???_1$ und $???_2$ einzusetzen?

3. Quiz

Sei $(z', B_1 \cdots B_k) \in \delta(z, \varepsilon, A)$. Dann gilt $???_1 \vdash ???_2$.
Was ist für $???_1$ und $???_2$ einzusetzen?

Antwort: (z, w, AW) und $(z', w, B_1 \cdots B_k W)$.

4. Quiz

Vervollständige: $w \in L(M)$ g.d.w. es existiert z, sodass $(z_0, w, \#) \vdash_M^*$??? für Kellerautomaten (die mit leerem Keller akzeptieren).

4. Quiz

Vervollständige: $w \in L(M)$ g.d.w. es existiert z, sodass $(z_0, w, \#) \vdash_M^* ???$ für Kellerautomaten (die mit leerem Keller akzeptieren).

Antwort: $(z, \varepsilon, \varepsilon)$.

5. Quiz

Vervollständige: $w \in L(M)$ g.d.w. es existieren z und W, sodass $(z_0, w, \#) \vdash_M^* ???$ für Kellerautomaten mit Endzuständen E.

5. Quiz

Vervollständige: $w \in L(M)$ g.d.w. es existieren z und W, sodass $(z_0, w, \#) \vdash_M^* ???$ für Kellerautomaten mit Endzuständen E.

Antwort: (z, ε, W), wo $z \in E$.

Aufgabe: PDA angeben

Geben Sie einen PDA an, der die Sprache $L = \{a^{2n}b^n \mid n \in \mathbb{N}_{>0}\}$ akzeptiert.

Aufgabe: PDA angeben

Geben Sie einen PDA an, der die Sprache $L = \{a^{2n}b^n \mid n \in \mathbb{N}_{>0}\}$ akzeptiert.

Antwort:

Aufgabe: PDA angeben

Geben Sie einen PDA an, der die Sprache $L = \{a^{2n}b^n \mid n \in \mathbb{N}_{>0}\}$ akzeptiert.

Antwort:

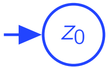
- $\underbrace{B \cdots B}_{i\text{-mal}}\#$ auf dem Keller heißt, noch i-mal b lesen.

Aufgabe: PDA angeben

Geben Sie einen PDA an, der die Sprache $L = \{a^{2n}b^n \mid n \in \mathbb{N}_{>0}\}$ akzeptiert.

Antwort:

- $\underbrace{B \cdots B}_{i\text{-mal}} \#$ auf dem Keller heißt, noch i-mal b lesen.

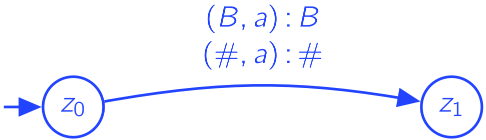


Aufgabe: PDA angeben

Geben Sie einen PDA an, der die Sprache $L = \{a^{2n}b^n \mid n \in \mathbb{N}_{>0}\}$ akzeptiert.

Antwort:

- $\underbrace{B \cdots B}_{i\text{-mal}}\#$ auf dem Keller heißt, noch i-mal b lesen.

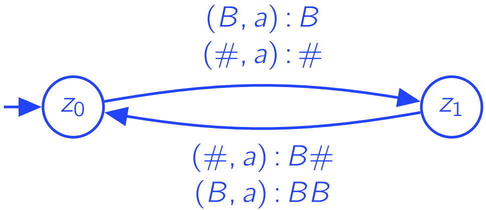


Aufgabe: PDA angeben

Geben Sie einen PDA an, der die Sprache $L = \{a^{2n}b^n \mid n \in \mathbb{N}_{>0}\}$ akzeptiert.

Antwort:

- $\underbrace{B \cdots B}_{i\text{-mal}}\#$ auf dem Keller heißt, noch i-mal b lesen.

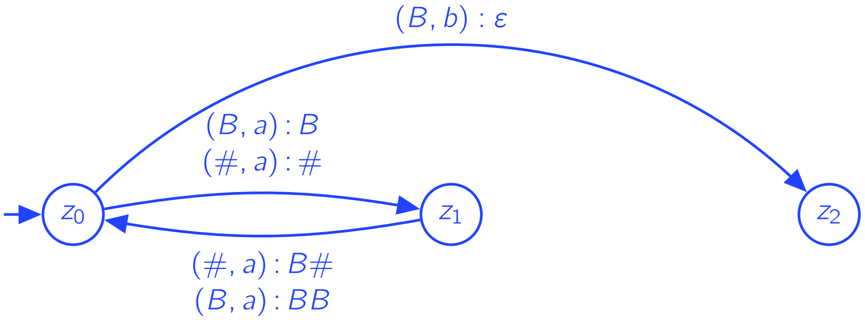


Aufgabe: PDA angeben

Geben Sie einen PDA an, der die Sprache $L = \{a^{2n}b^n \mid n \in \mathbb{N}_{>0}\}$ akzeptiert.

Antwort:

- $\underbrace{B \cdots B}_{i\text{-mal}}\#$ auf dem Keller heißt, noch i-mal b lesen.

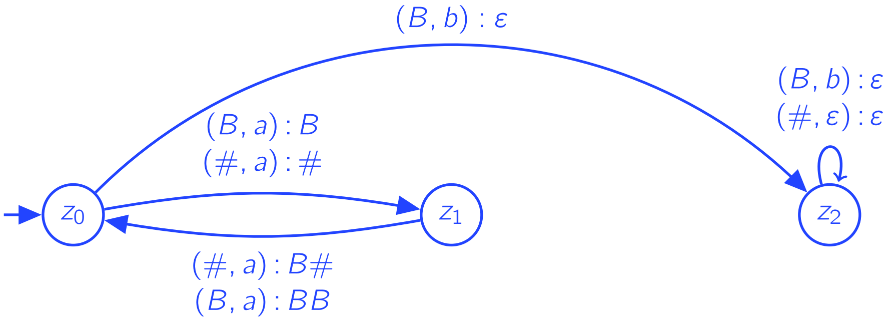


Aufgabe: PDA angeben

Geben Sie einen PDA an, der die Sprache $L = \{a^{2n}b^n \mid n \in \mathbb{N}_{>0}\}$ akzeptiert.

Antwort:

- $\underbrace{B \cdots B}_{i\text{-mal}}\#$ auf dem Keller heißt, noch i-mal b lesen.



Quiz

Betrachte einen deterministischen Kellerautomaten (mit Endzuständen). Welche der folgenden Bedingungen gelten stets?

a) Es gibt keine ε-Übergänge.

b) Wenn es einen ε-Übergang mit Kellersymbol A und Zustand z gibt, dann gibt es keinen anderen Übergang für A und z.

c) Wenn $(z', W) \in \delta(z, a, A)$ gilt, dann gibt es keinen ε-Übergang für Zustand z und Kellersymbol A.

d) Für alle z, a, A gilt $\delta(z, a, A) \neq \emptyset$.

e) Wenn $(z', W) \in \delta(z, a, A)$ gilt, dann gibt es keinen Zustand und Kellersymbole $(z'', W') \in \delta(z, a, A')$ für alle $A' \neq A$.

Quiz

Betrachte einen deterministischen Kellerautomaten (mit Endzuständen). Welche der folgenden Bedingungen gelten stets?

a) Es gibt keine ε-Übergänge.

b) Wenn es einen ε-Übergang mit Kellersymbol A und Zustand z gibt, dann gibt es keinen anderen Übergang für A und z.

c) Wenn $(z', W) \in \delta(z, a, A)$ gilt, dann gibt es keinen ε-Übergang für Zustand z und Kellersymbol A.

d) Für alle z, a, A gilt $\delta(z, a, A) \neq \emptyset$.

e) Wenn $(z', W) \in \delta(z, a, A)$ gilt, dann gibt es keinen Zustand und Kellersymbole $(z'', W') \in \delta(z, a, A')$ für alle $A' \neq A$.

Antwort: b) und c).
Allgemein ist die Bedingung bei DPDAs $|\delta(z, a, A)| + |\delta(z, \varepsilon, A)| \leq 1$.

3. Turingmaschinen

Definition einer Turingmaschine

Definition

Eine Turingmaschine (TM) ist ein 7-Tupel $M = (Z, \Sigma, \Gamma, \delta, z_0, \Box, E)$, wobei:

- Z ist eine endliche Menge von Zuständen
- Σ ist das (endliche) Eingabealphabet
- $\Gamma \supset \Sigma$ ist das (endliche) Bandalphabet
- δ ist die Überführungsfunktion
  - für deterministische TM (DTM): $\delta : (Z \setminus E) \times \Gamma \to Z \times \Gamma \times \{L, R, N\}$
  - für nichtdeterministische TM (NTM): $\delta : (Z \setminus E) \times \Gamma \to \mathcal{P}(Z \times \Gamma \times \{L, R, N\})$
- $z_0 \in Z$ ist der Startzustand
- $\Box \in \Gamma \setminus \Sigma$ ist das Blank-Symbol
- $E \subseteq Z$ ist die Menge der Endzustände.

Konfigurationen

Definition

Sei $M = (Z, \Sigma, \Gamma, \delta, z_0, \Box, E)$ eine Turingmaschine.
Eine Konfiguration von M ist ein Wort $wzw' \in \Gamma^* Z \Gamma^*$, wobei:

- z ist der aktuelle Zustand von M
- $\cdots \Box\Box ww' \Box\Box \cdots$ steht auf dem Band
- der Schreib-Lesekopf steht auf dem ersten Symbol von w'.

Konfigurationen

Definition

Sei $M = (Z, \Sigma, \Gamma, \delta, z_0, \square, E)$ eine Turingmaschine.
Eine Konfiguration von M ist ein Wort $wzw' \in \Gamma^* Z \Gamma^*$, wobei:

- z ist der aktuelle Zustand von M
- $\cdots \square\square ww' \square\square \cdots$ steht auf dem Band
- der Schreib-Lesekopf steht auf dem ersten Symbol von w'.

Definition

Sei $M = (Z, \Sigma, \Gamma, \delta, z_0, \square, E)$ eine Turingmaschine.
Für ein Eingabewort w ist die Startkonfiguration $Start_M(w)$ von M das Wort $z_0 w$.
Im Spezialfall $w = \varepsilon$ ist die Startkonfiguration $z_0 \square$.

1. Quiz

Sei $\delta(z_0, a) = (z_1, A, R)$, $\delta(z_0, b) = (z_1, B, L)$ und $\delta(z_0, c) = (z_1, C, N)$.
Was ist die Nachfolgekonfiguration von abz_0cabc?

a) $abCz_0abcC$

b) abz_1Cabc

c) az_1Bcabc

d) $z_1AbcAbc$

1. Quiz

Sei $\delta(z_0, a) = (z_1, A, R)$, $\delta(z_0, b) = (z_1, B, L)$ und $\delta(z_0, c) = (z_1, C, N)$.
Was ist die Nachfolgekonfiguration von abz_0cabc?

a) $abCz_0abcC$

b) abz_1Cabc

c) az_1Bcabc

d) $z_1AbcAbc$

Antwort: b).

2. Quiz

Sei $\delta(z_0, a) = (z_1, A, R)$, $\delta(z_0, b) = (z_1, B, L)$ und $\delta(z_0, c) = (z_1, C, N)$.
Was ist die Nachfolgekonfiguration von $z_0 bcbca$?

a) $Bz_1 cbca$

b) $z_1 Bcbca$

c) $Bcbcaz_1$

d) $z_1 \square Bcbca$

2. Quiz

Sei $\delta(z_0, a) = (z_1, A, R)$, $\delta(z_0, b) = (z_1, B, L)$ und $\delta(z_0, c) = (z_1, C, N)$.
Was ist die Nachfolgekonfiguration von $z_0 bcbca$?

a) $Bz_1 cbca$

b) $z_1 Bcbca$

c) $Bcbcaz_1$

d) $z_1 \square Bcbca$

Antwort: d).

Akzeptierte Sprache einer Turingmaschine

Definition

Sei $M = (Z, \Sigma, \Gamma, \delta, z_0, \Box, E)$ eine Turingmaschine.
Die von M akzeptierte Sprache $L(M)$ ist definiert als

$$L(M) := \{w \in \Sigma^* \mid Start_M(w) \vdash_M^* uzv \text{ für } u, v \in \Gamma^*,\ z \in E\}$$

Weitere Begriffe:

- Die TM akzeptiert heißt, die TM erreicht einen Endzustand.
- Die TM verwirft heißt, die TM erreicht keinen Endzustand.

Vorgehen um eine Turingmaschine anzugeben

Schritte:

1. Überlege Grundgedanke zum Erkennen von der gegebenen Sprache.
2. „Programmiere“ dies erstmal grob, phasenweise.
3. Modelliere es genauer mit Zuständen und Übergängen.

Aufgabe: Turingmaschine angeben

Geben Sie eine Turingmaschine an, die genau bei Eingabe a^{2^n} mit $n \in \mathbb{N}$ akzeptiert.

Aufgabe: Turingmaschine angeben

Geben Sie eine Turingmaschine an, die genau bei Eingabe a^{2^n} mit $n \in \mathbb{N}$ akzeptiert.

Antwort:

1. Überlege Grundgedanke:
   - Auf dem Band steht zu Beginn $\cdots \Box a \cdots a \Box \cdots$.
   - Erkenne, ob Anzahl an a's eine Zahl der Form 2^n ist.
   - Es gilt $2^n = 2 \cdot 2^{n-1}$. Streicht man die Hälfte aller a's, so muss der Rest immer noch von der Form a^{2^m} sein.
   - $a^{2^n} \rightsquigarrow a^{2^{n-1}} \rightsquigarrow a^{2^{n-2}} \rightsquigarrow \cdots$
   - Wenn man keine gleichen Hälften hat, dann stand dort a^n mit n ungerade. Verwirf.
   - Ausnahmefall: $n = 0$: Auf den Band steht ein a.

Aufgabe: Turingmaschine angeben

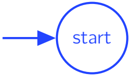
2. „Programmiere“:
   - Wenn nur ein a, dann akzeptiere.
   - Hälfte streichen: Ersetze jedes zweite a durch x.
   - Wenn ungerade viele a’s und mindestens ein a gestrichen wurden, dann verwirf.
   - Starte von Neuem mit nächster Phase.

Aufgabe: Turingmaschine angeben

3. Modelliere:

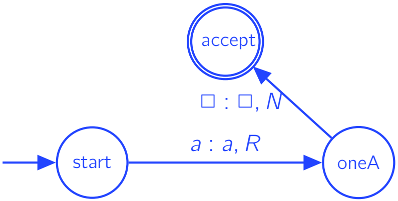
Aufgabe: Turingmaschine angeben

3. Modelliere:



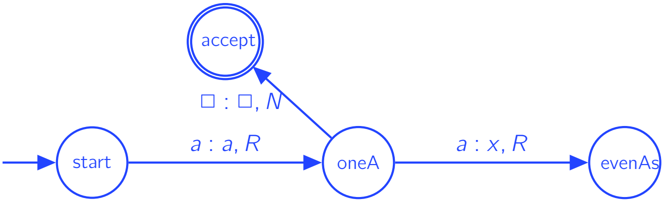
Aufgabe: Turingmaschine angeben

3. Modelliere:



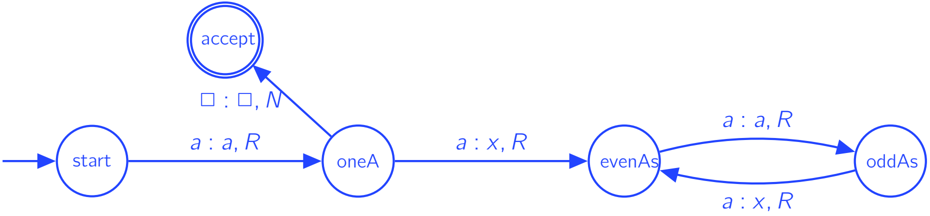
Aufgabe: Turingmaschine angeben

3. Modelliere:



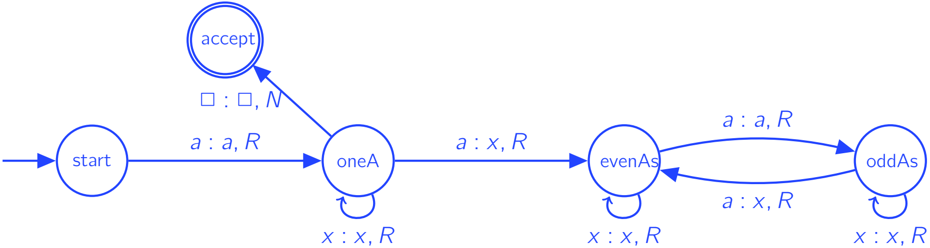
Aufgabe: Turingmaschine angeben

3. Modelliere:



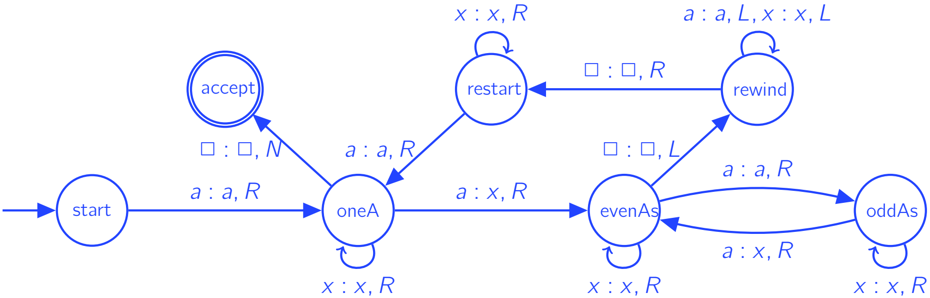
Aufgabe: Turingmaschine angeben

3. Modelliere:



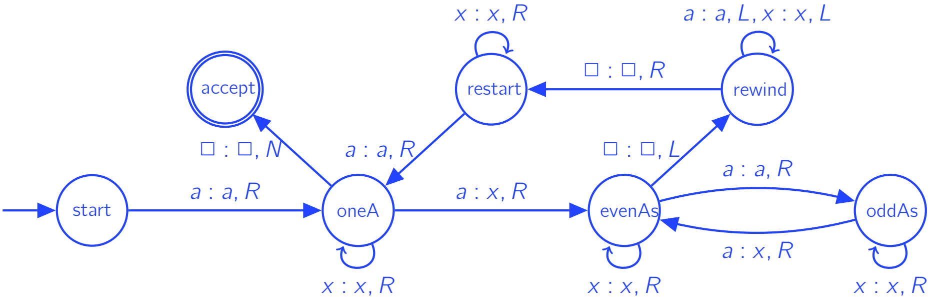
Aufgabe: Turingmaschine angeben

3. Modelliere:



Aufgabe: Turingmaschine angeben

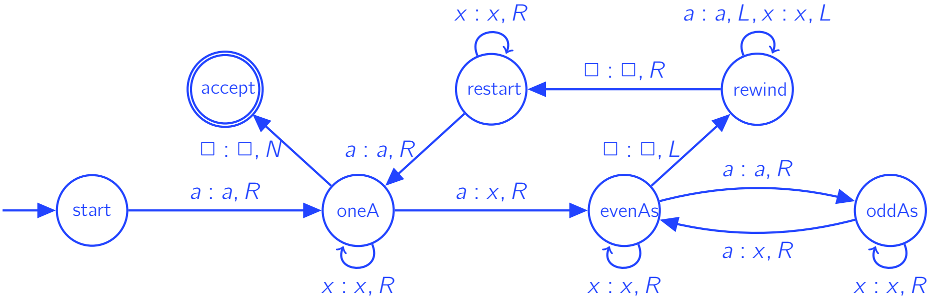
3. Modelliere:



Für DTM bräuchten wir auch einen Müllzustand und dazugehörige Übergänge.

Aufgabe: Turingmaschine angeben

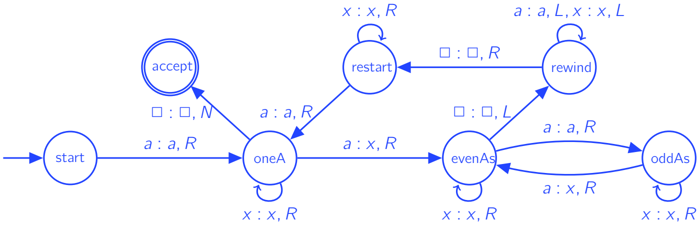
3. Modelliere:



Für DTM bräuchten wir auch einen Müllzustand und dazugehörige Übergänge.

`http://turingmachinesimulator.com/shared/fsxyubfqam`

Aufgabe: Turingmaschine angeben



Lauf der Turingmaschine auf $aaaa$:

start $aaaa$	$\vdash$ a oneA aaa	$\vdash$ ax evenAs aa	$\vdash$ axa oddAs a
$\vdash$ $axax$ evenAs $\square$	$\vdash$ axa rewind $x\square$	$\vdash$ ax rewind $ax\square$	$\vdash$ a rewind $xax\square$
$\vdash$ rewind $axax\square$	$\vdash$ rewind $\square axax\square$	$\vdash$ $\square$ restart $axax\square$	$\vdash$ $\square a$ oneA $xax\square$
$\vdash$ $\square ax$ oneA $ax\square$	$\vdash$ $\square axx$ evenAs $x\square$	$\vdash$ $\square axxx$ evenAs $\square$	$\vdash$ $\square axx$ rewind $x\square$
$\vdash$ $\square ax$ rewind $xx\square$	$\vdash$ $\square a$ rewind $xxx\square$	$\vdash$ $\square$ rewind $axxx\square$	$\vdash$ rewind $\square axxx\square$
$\vdash$ $\square$restart $axxx\square$	$\vdash$ $\square a$ oneA $xxx\square$	$\vdash$ $\square ax$ oneA $xx\square$	$\vdash$ $\square axx$ oneA $x\square$
$\vdash$ $\square axxx$ oneA $\square$	$\vdash$ $\square axxx$ accept $\square$		