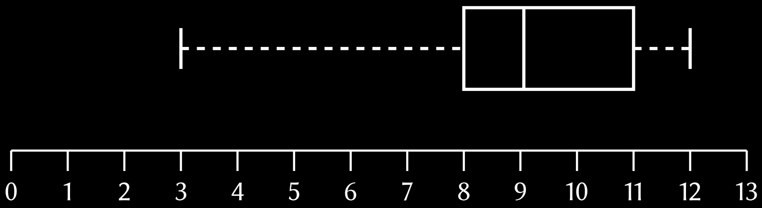
Logische Complexiteit

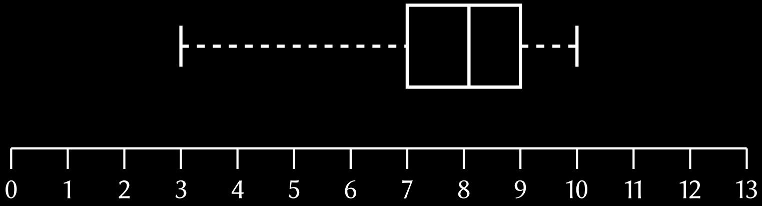
V : Greibach Normaalvorm

Jeroen Goudsmit
Universiteit Utrecht
dinsdag 21 februari 2012

Inleveropgave 1



Inleveropgave 1



Inleveropgave 2.2

Lemma
Stel dat $\mathcal{L}_1$ en $\mathcal{L}_2$ reguliere talen zijn. Nu is $\mathcal{L}_1 - \mathcal{L}_2$ ook regulier.

Inleveropgave 2.2

Lemma
Stel dat $\mathcal{L}_1$ en $\mathcal{L}_2$ reguliere talen zijn. Nu is $\mathcal{L}_1 - \mathcal{L}_2$ ook regulier.

Bewijs.

Zij $\mathcal{L}_1$ en $\mathcal{L}_2$ reguliere talen.

Inleveropgave 2.2

Lemma

Stel dat $\mathcal{L}_1$ en $\mathcal{L}_2$ reguliere talen zijn. Nu is $\mathcal{L}_1 - \mathcal{L}_2$ ook regulier.

Bewijs.

Zij $\mathcal{L}_1$ en $\mathcal{L}_2$ reguliere talen. Er geldt dat

$$\mathcal{L}_1 - \mathcal{L}_2 = \mathcal{L}_1 \cap (\Sigma^* - \mathcal{L}_2) = \mathcal{L}_1 \cap \overline{\mathcal{L}_2}. \tag{1}$$

Inleveropgave 2.2

Lemma
Stel dat $\mathcal{L}_1$ en $\mathcal{L}_2$ reguliere talen zijn. Nu is $\mathcal{L}_1 - \mathcal{L}_2$ ook regulier.

Bewijs.

Zij $\mathcal{L}_1$ en $\mathcal{L}_2$ reguliere talen. Er geldt dat

$$\mathcal{L}_1 - \mathcal{L}_2 = \mathcal{L}_1 \cap (\Sigma^* - \mathcal{L}_2) = \mathcal{L}_1 \cap \overline{\mathcal{L}_2}. \tag{1}$$

Reguliere talen zijn gesloten onder complement, dus $\overline{\mathcal{L}_2}$ is regulier.

Inleveropgave 2.2

Lemma
Stel dat $\mathcal{L}_1$ en $\mathcal{L}_2$ reguliere talen zijn. Nu is $\mathcal{L}_1 - \mathcal{L}_2$ ook regulier.

Bewijs.

Zij $\mathcal{L}_1$ en $\mathcal{L}_2$ reguliere talen. Er geldt dat

$$\mathcal{L}_1 - \mathcal{L}_2 = \mathcal{L}_1 \cap (\Sigma^* - \mathcal{L}_2) = \mathcal{L}_1 \cap \overline{\mathcal{L}_2}. \tag{1}$$

Reguliere talen zijn gesloten onder complement, dus $\overline{\mathcal{L}_2}$ is regulier. Evenzo zijn ze gesloten onder doorsnede, dus $\mathcal{L}_1 \cap \overline{\mathcal{L}_2}$ is ook regulier.

Inleveropgave 2.2

Lemma
Stel dat $\mathcal{L}_1$ en $\mathcal{L}_2$ reguliere talen zijn. Nu is $\mathcal{L}_1 - \mathcal{L}_2$ ook regulier.

Bewijs.

Zij $\mathcal{L}_1$ en $\mathcal{L}_2$ reguliere talen. Er geldt dat

$$\mathcal{L}_1 - \mathcal{L}_2 = \mathcal{L}_1 \cap (\Sigma^* - \mathcal{L}_2) = \mathcal{L}_1 \cap \overline{\mathcal{L}_2}. \tag{1}$$

Reguliere talen zijn gesloten onder complement, dus $\overline{\mathcal{L}_2}$ is regulier. Evenzo zijn ze gesloten onder doorsnede, dus $\mathcal{L}_1 \cap \overline{\mathcal{L}_2}$ is ook regulier. Dit bewijst het gewenste wegens (1). ■

Inleveropgave 2.3.a

Lemma
De taal $\mathcal{L} = \left\{ w \in \Sigma^* \;\middle|\; w \text{ bevat twee keer zoveel a's als b's} \right\}$ *is niet regulier.*

Inleveropgave 2.3.a

Lemma
De taal $\mathcal{L} = \{ w \in \Sigma^* \mid w \text{ bevat twee keer zoveel a's als b's} \}$ is niet regulier.

Bewijs.

Stel $\mathcal{L}$ was wel regulier.

Inleveropgave 2.3.a

Lemma
De taal $\mathcal{L} = \{ w \in \Sigma^* \mid w \text{ bevat twee keer zoveel a's als b's} \}$ *is niet regulier.*

Bewijs.

Stel $\mathcal{L}$ was wel regulier. Wegens het pomplemma was er dan een pomplengte p.

Inleveropgave 2.3.a

Lemma
De taal $\mathcal{L} = \{ w \in \Sigma^* \mid w \text{ bevat twee keer zoveel a's als b's} \}$ is niet regulier.

Bewijs.

Stel $\mathcal{L}$ was wel regulier. Wegens het pomplemma was er dan een pomplengte p. Bekijk het woord $w = a^{2p}b^p$ en zie dat $|w| \geq p$ en $w \in \mathcal{L}$.

Inleveropgave 2.3.a

Lemma
De taal $\mathcal{L} = \left\{ w \in \Sigma^* \;\middle|\; w \text{ bevat twee keer zoveel a's als b's} \right\}$ *is niet regulier.*

Bewijs.

Stel $\mathcal{L}$ was wel regulier. Wegens het pomplemma was er dan een pomplengte p. Bekijk het woord $w = a^{2p}b^p$ en zie dat $|w| \geq p$ en $w \in \mathcal{L}$. Zij $x, y, z \in \Sigma^*$ zó dat $w = xyz$, $|xy| \leq p$ en $|y| \geq 1$.

Inleveropgave 2.3.a

Lemma

De taal $\mathcal{L} = \{ w \in \Sigma^* \mid w \text{ bevat twee keer zoveel a's als b's} \}$ is niet regulier.

Bewijs.

Stel $\mathcal{L}$ was wel regulier. Wegens het pomplemma was er dan een pomplengte p. Bekijk het woord $w = a^{2p}b^p$ en zie dat $|w| \geq p$ en $w \in \mathcal{L}$. Zij $x, y, z \in \Sigma^*$ zó d $w = xyz$, $|xy| \leq p$ en $|y| \geq 1$. Wegens het voorgaande geldt nu $x = a^k$, $y = a^l$ en $z = a^m b^p$ voor zekere $k, l, m \in \mathbb{N}$ met $k + l + m = 2p$.

Inleveropgave 2.3.a

Lemma
De taal $\mathcal{L} = \{ w \in \Sigma^* \mid w \text{ bevat twee keer zoveel a's als b's} \}$ *is niet regulier.*

Bewijs.

Stel $\mathcal{L}$ was wel regulier. Wegens het pomplemma was er dan een pomplengte p. Bekijk het woord $w = a^{2p}b^p$ en zie dat $|w| > p$ en $w \in \mathcal{L}$. Zij $x, y, z \in \Sigma^*$ zó dat $w = xyz$, $|xy| \leq p$ er $|y| \geq 1$. Wegens het voorgaande geldt nu $x = a^k$, $y = a^l$ en $z = a^m b^p$ voor zekere $k, l, m \in \mathbb{N}$ met $k + l + m = 2p$. Stel nu dat $xz = a^{k+m}b^p \in \mathcal{L}$, dan moet gelden $k + m = 2p$. Dus $l = 0$, tegenspraak.

Inleveropgave 2.3.a

Lemma
De taal $\mathcal{L} = \{ w \in \Sigma^* \mid w \text{ bevat twee keer zoveel a's als b's} \}$ *is niet regulier.*

Bewijs.

Stel $\mathcal{L}$ was wel regulier. Wegens het pomplemma was er dan een pomplengte p. Bekijk het woord $w = a^{2p}b^p$ en zie dat $|w| \geq p$ en $w \in \mathcal{L}$. Zij $x, y, z \in \Sigma^*$ zó dat $w = xyz, |xy| \leq p$ en $|y| \geq 1$. Wegens het voorgaande geldt nu $x = a^k$, $y = a^l$ en $z = a^m b^p$ voor zekere $k, l, m \in \mathbb{N}$ met $k + l + m = 2p$. Stel nu dat $xz = a^{k+m}b^p \in \mathcal{L}$, dan moet gelden $k + m = 2p$. Dus $l = 0$, tegenspraak. Hiermee geldt $xy^0z \notin \mathcal{L}$, dus er is geen pomplengte, dus $\mathcal{L}$ is niet regulier. ■

Chomsky Normaalvorm

Elke CFG past in Chomsky Normaalvorm

Chomsky Normaalvorm

$\langle V, \Sigma, R, S \rangle$

$$\begin{array}{ll} A \rightarrow B\ C & B, C \in V - S \\ A \rightarrow x & x \in \Sigma \\ S \rightarrow \varepsilon & \end{array}$$

Elke CFG past in Chomsky Normaalvorm

Chomsky Normaalvorm

$\langle V, \Sigma, R, S \rangle$

$A \rightarrow B\ C \qquad B, C \in V - S$

$A \rightarrow x \qquad x \in \Sigma$

$S \rightarrow \varepsilon$

Elke CFG past in Chomsky Normaalvorm

verwijder lege regels → verwijder enkele regels → ontkoppel ketens

Greibach Normaalvorm

Elke CFG past in Greibach Normaalvorm

Greibach Normaalvorm

$\langle V, \Sigma, R, \mathrm{S} \rangle$

$\mathrm{A} \to x\ \underline{\mathrm{B}} \quad x \in \Sigma$ en $\underline{\mathrm{B}} \in \left(V - \{\mathrm{S}\}\right)^*$
$\mathrm{S} \to \varepsilon$

Elke CFG past in Greibach Normaalvorm

Voorbeeld

Grammatica

$$
\begin{array}{lcccc}
S & \rightarrow & a\ S\ A & \mid & b\ S\ B \\
 & \mid & a\ A & \mid & b\ B \\
A & \rightarrow & a & & \\
B & \rightarrow & b & &
\end{array}
$$

Voorbeeld

Grammatica

$$
\begin{array}{lcccc}
S & \rightarrow & a\ S\ A & \mid & b\ S\ B \\
 & \mid & a\ A & \mid & b\ B \\
A & \rightarrow & a & & \\
B & \rightarrow & b & &
\end{array}
$$

Lezen

S

Voorbeeld

Grammatica

$$\begin{array}{rcccc} S & \rightarrow & a\ S\ A & \mid & b\ S\ B \\ & \mid & a\ A & \mid & b\ B \\ A & \rightarrow & a & & \\ B & \rightarrow & b & & \end{array}$$

Lezen

S

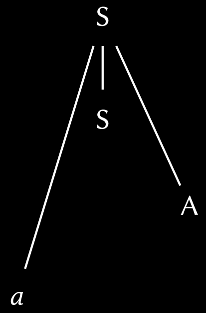
a

Voorbeeld

Grammatica

$$
\begin{array}{lcccc}
S & \rightarrow & a\ S\ A & | & b\ S\ B \\
 & | & a\ A & | & b\ B \\
A & \rightarrow & a & & \\
B & \rightarrow & b & &
\end{array}
$$

Lezen

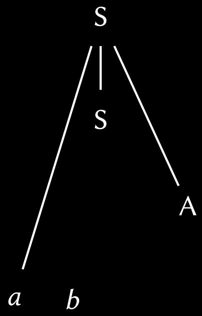


Voorbeeld

Grammatica

$$
\begin{array}{lcccc}
\mathrm{S} & \rightarrow & a\ \mathrm{S}\ A & | & b\ \mathrm{S}\ B \\
 & | & a\ A & | & b\ B \\
A & \rightarrow & a & & \\
B & \rightarrow & b & &
\end{array}
$$

Lezen

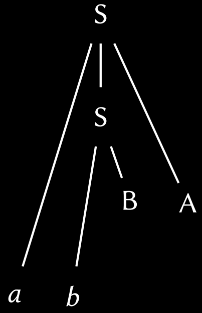


Voorbeeld

Grammatica

$$
\begin{array}{lcccc}
\mathrm{S} & \rightarrow & a\ \mathrm{S}\ A & | & b\ \mathrm{S}\ B \\
 & | & a\ A & | & b\ B \\
A & \rightarrow & a & & \\
B & \rightarrow & b & &
\end{array}
$$

Lezen

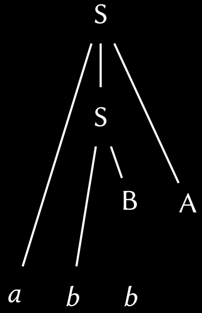


Voorbeeld

Grammatica

$$
\begin{array}{lcccc}
S & \rightarrow & a\ S\ A & \mid & b\ S\ B \\
 & \mid & a\ A & \mid & b\ B \\
A & \rightarrow & a & & \\
B & \rightarrow & b & &
\end{array}
$$

Lezen

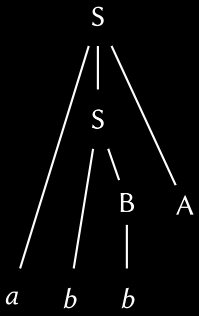


Voorbeeld

Grammatica

$$
\begin{array}{lclcl}
S & \rightarrow & a\ S\ A & \mid & b\ S\ B \\
 & \mid & a\ A & \mid & b\ B \\
A & \rightarrow & a & & \\
B & \rightarrow & b & &
\end{array}
$$

Lezen

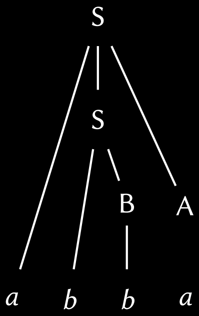


Voorbeeld

Grammatica

$$
\begin{array}{lclcl}
S & \rightarrow & a\ S\ A & | & b\ S\ B \\
 & | & a\ A & | & b\ B \\
A & \rightarrow & a & & \\
B & \rightarrow & b & &
\end{array}
$$

Lezen

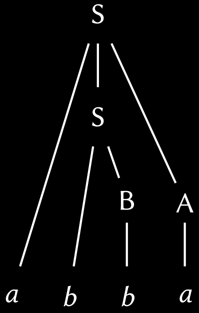


Voorbeeld

Grammatica

$$
\begin{array}{cccccc}
\mathrm{S} & \rightarrow & a\ \mathrm{S}\ A & | & b\ \mathrm{S}\ B \\
 & | & a\ A & | & b\ B \\
A & \rightarrow & a & & \\
B & \rightarrow & b & &
\end{array}
$$

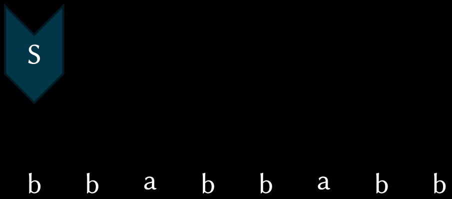
Lezen



Voorbeeld Reloaded

b b a b b a b b

Voorbeeld Reloaded



Voorbeeld Reloaded

b b a b b a b b

Voorbeeld Reloaded

b b a b b a b b

Voorbeeld Reloaded

b b a b b a b b

Voorbeeld Reloaded

b b a b b a b b

Voorbeeld Reloaded

b b a b b a b b

Voorbeeld Reloaded

b b a b b a b b

Voorbeeld Reloaded

B

b b a b b a b b

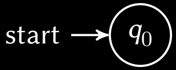
Voorbeeld Reloaded

b b a b b a b b

Model van Berekening

Grammatica

$$
\begin{aligned}
S &\rightarrow a\ S\ A \\
S &\rightarrow b\ S\ B \\
S &\rightarrow a\ A \\
S &\rightarrow b\ B \\
A &\rightarrow a \\
B &\rightarrow b
\end{aligned}
$$



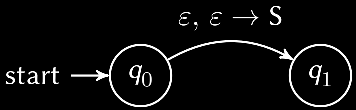
Machine

Model van Berekening

Grammatica

$$
\begin{aligned}
S &\rightarrow a\ S\ A \\
S &\rightarrow b\ S\ B \\
S &\rightarrow a\ A \\
S &\rightarrow b\ B \\
A &\rightarrow a \\
B &\rightarrow b
\end{aligned}
$$

Machine

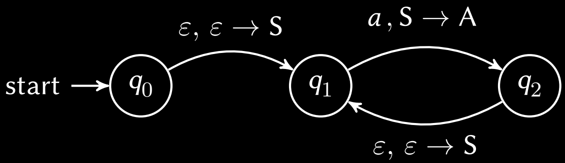


Model van Berekening

Grammatica

$$
\begin{aligned}
S &\rightarrow a\ S\ A \\
S &\rightarrow b\ S\ B \\
S &\rightarrow a\ A \\
S &\rightarrow b\ B \\
A &\rightarrow a \\
B &\rightarrow b
\end{aligned}
$$

Machine

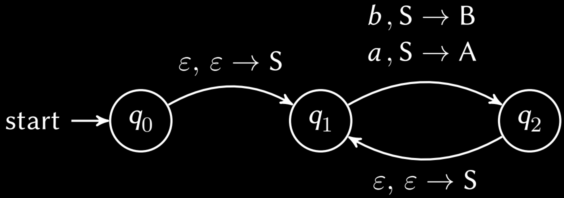


Model van Berekening

Grammatica

$$
\begin{aligned}
S &\rightarrow a\ S\ A \\
S &\rightarrow b\ S\ B \\
S &\rightarrow a\ A \\
S &\rightarrow b\ B \\
A &\rightarrow a \\
B &\rightarrow b
\end{aligned}
$$

Machine

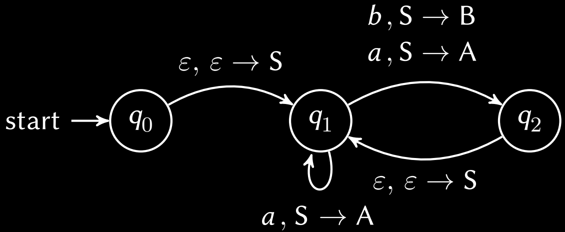


Model van Berekening

Grammatica

$$
\begin{array}{lcl}
\mathsf{S} & \rightarrow & a\ \mathsf{S}\ \mathsf{A} \\
\mathsf{S} & \rightarrow & b\ \mathsf{S}\ \mathsf{B} \\
\mathsf{S} & \rightarrow & a\ \mathsf{A} \\
\mathsf{S} & \rightarrow & b\ \mathsf{B} \\
\mathsf{A} & \rightarrow & a \\
\mathsf{B} & \rightarrow & b
\end{array}
$$

Machine

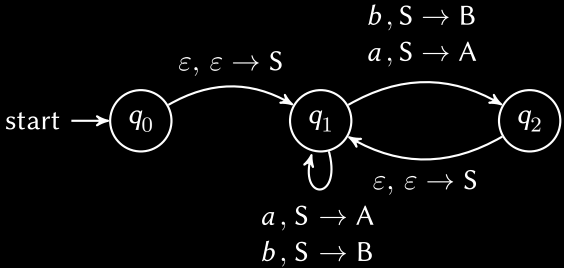


Model van Berekening

Grammatica

$$
\begin{array}{lcl}
S & \rightarrow & a\ S\ A \\
S & \rightarrow & b\ S\ B \\
S & \rightarrow & a\ A \\
S & \rightarrow & b\ B \\
A & \rightarrow & a \\
B & \rightarrow & b
\end{array}
$$

Machine

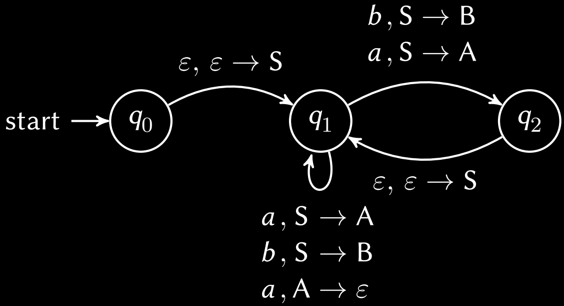


Model van Berekening

Grammatica

$$
\begin{aligned}
S &\rightarrow a\,S\,A \\
S &\rightarrow b\,S\,B \\
S &\rightarrow a\,A \\
S &\rightarrow b\,B \\
A &\rightarrow a \\
B &\rightarrow b
\end{aligned}
$$

Machine

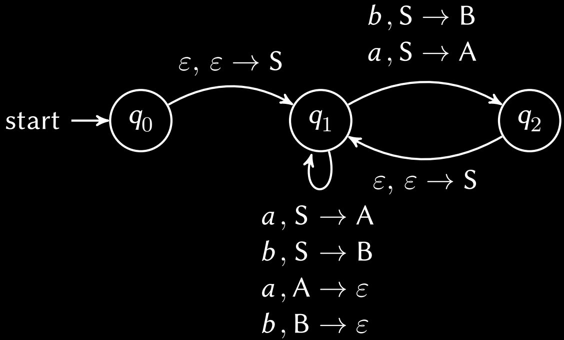


Model van Berekening

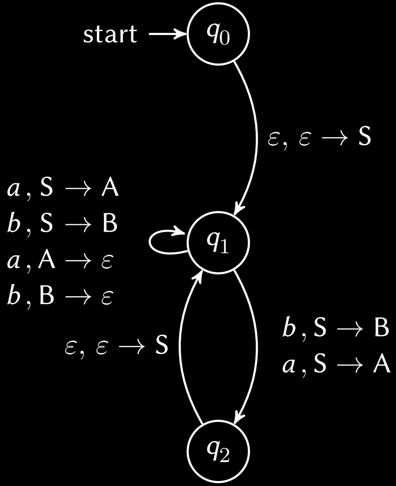
Grammatica

$$
\begin{array}{lcl}
S & \rightarrow & a\ S\ A \\
S & \rightarrow & b\ S\ B \\
S & \rightarrow & a\ A \\
S & \rightarrow & b\ B \\
A & \rightarrow & a \\
B & \rightarrow & b
\end{array}
$$

Machine

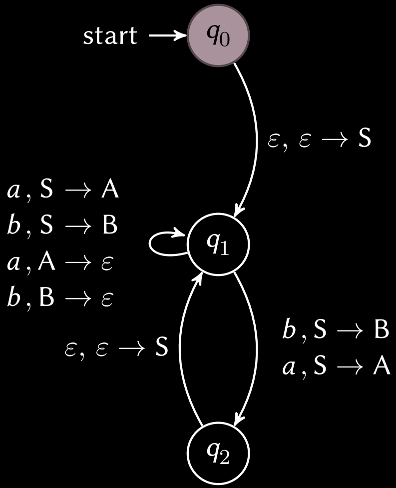


Return of the Voorbeeld



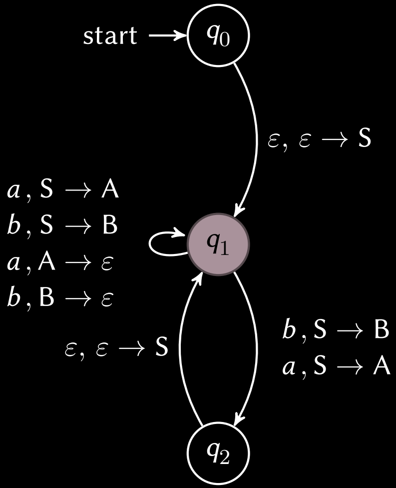
b b a a b b

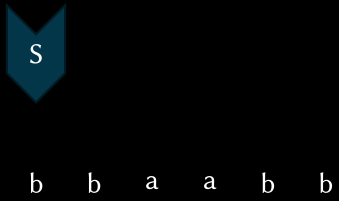
Return of the Voorbeeld



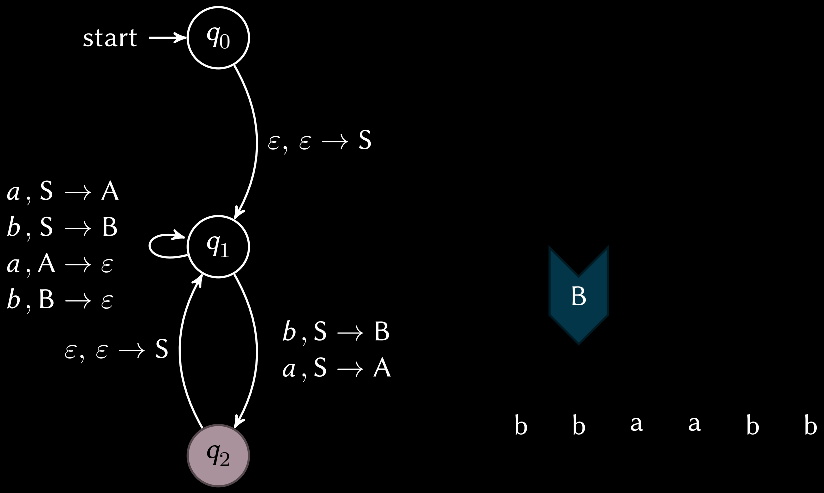
b b a a b b

Return of the Voorbeeld

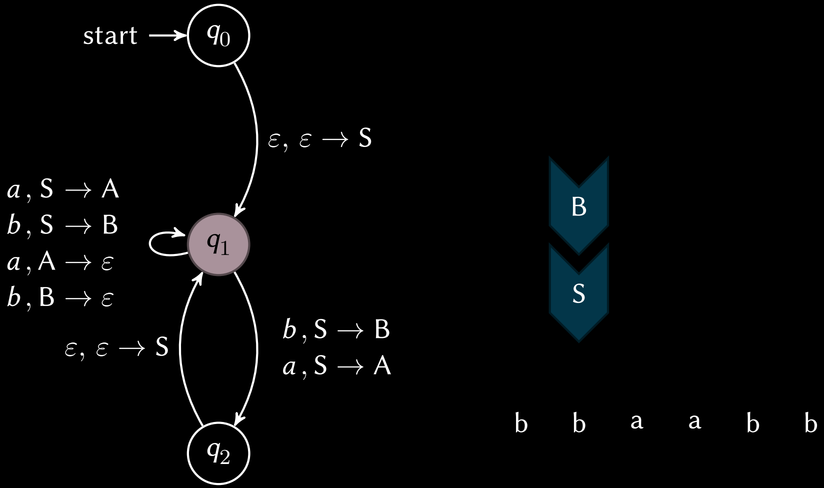




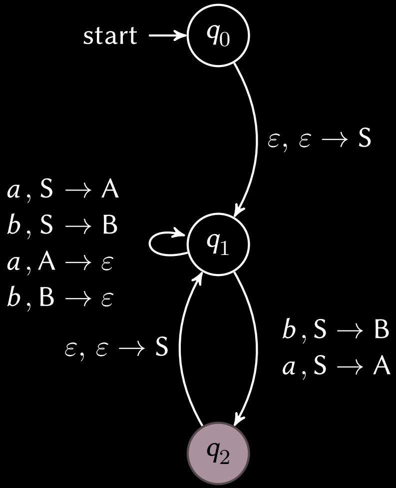
Return of the Voorbeeld



Return of the Voorbeeld

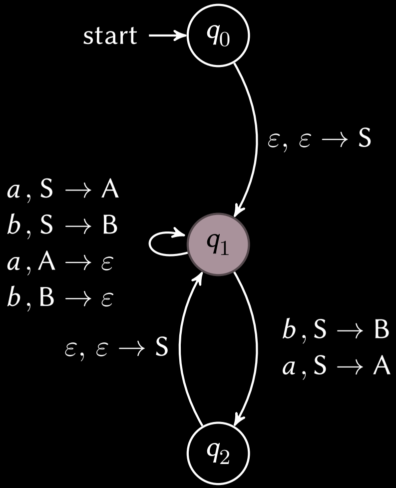


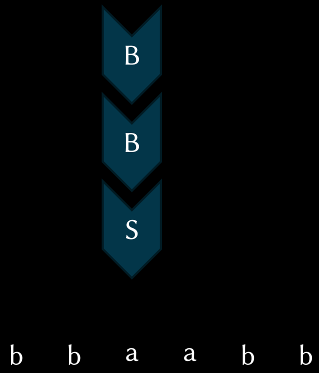
Return of the Voorbeeld



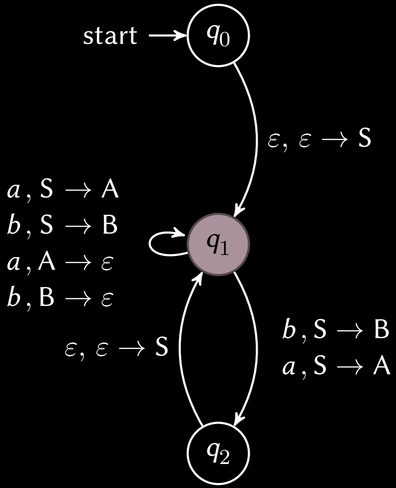
b b a a b b

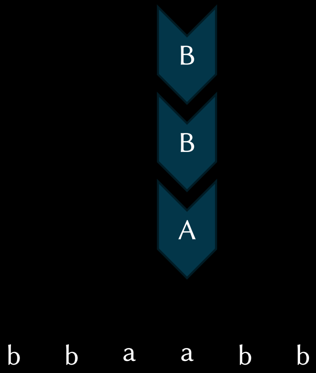
Return of the Voorbeeld



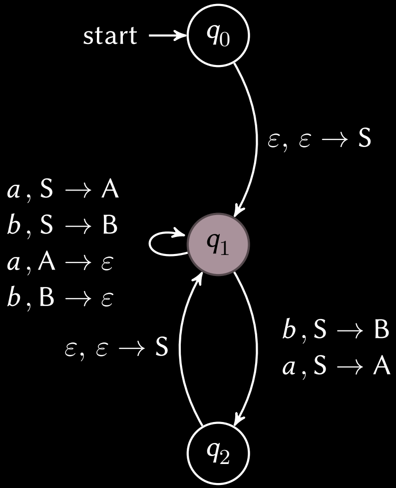


Return of the Voorbeeld



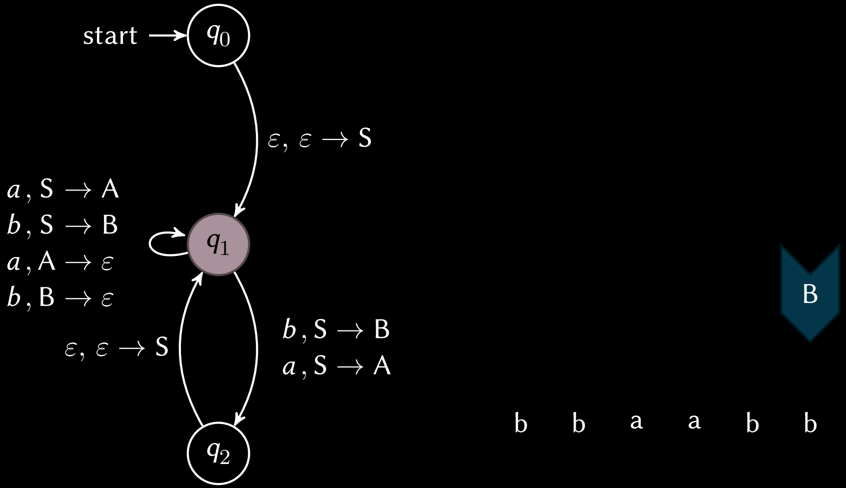


Return of the Voorbeeld

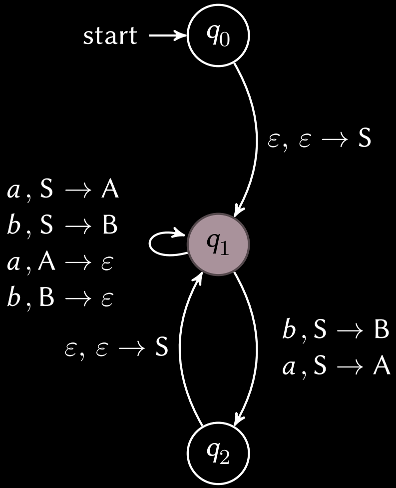


b b a a b b

Return of the Voorbeeld



Return of the Voorbeeld



b b a a b b

PDA

Definitie

Een PDA is een tupel $\langle Q, \Sigma, \Gamma, \delta, q_0, F \rangle$ met:

Q	eindige verzameling	toestanden
Σ	eindige verzameling	alfabet
Γ	eindige verzameling	stack alfabet
$\delta : Q \times \Sigma_\varepsilon \times \Gamma_\varepsilon \to \mathbf{P}\,(Q \times \Gamma_\varepsilon)$		transitiefunctie
$q_0 \in Q$		begintoestand
$F \subseteq Q$		eindtoestanden

Berekening

Definitie
Een berekening van $w \in \Sigma^*$ op M bestaat uit

$$v_1 \dots v_n \in \Sigma_\varepsilon^* \quad r_0, \dots, r_n \in Q \quad u_0, \dots, u_n \in \Gamma^*$$

zodat $\langle r_{i+1}, y \rangle \in \delta(r_i, v_{i+1}, x)$ voor $u_i = xu$ en $u_{i+1} = yu$ met $x, y \in \Gamma_\varepsilon$ en $u \in \Gamma^*$.

Berekening

$\langle\, Q, \Sigma, \Gamma, \delta, q_0, F \,\rangle$

Definitie
Een berekening van $w \in \Sigma^*$ op M bestaat uit

$$\underbrace{v_1 \ldots v_n}_{=\, w} \in \Sigma_\varepsilon^* \qquad \underbrace{r_0}_{=\, q_0}, \ldots, r_n \in Q \qquad \underbrace{u_0}_{=\, \varepsilon}, \ldots, u_n \in \Gamma^*$$

zodat $\langle\, r_{i+1}, y \rangle \in \delta\big(r_i, v_{i+1}, x\big)$ voor $u_i = xu$ en $u_{i+1} = yu$ met $x, y \in \Gamma_\varepsilon$ en $u \in \Gamma^*$. Dit accepteert als $r_n \in F$.

Berekening

$\vdash$ relatie op $Q \times \Sigma^* \times \Gamma^*$

Berekening

$\vdash$ relatie op $Q \times \Sigma^* \times \Gamma^*$

$$\frac{\langle r, z \rangle \in \delta(q, x, y)}{q, x\ w, y\ v \vdash r, w, z\ v}$$

Berekening

$\vdash$ relatie op $Q \times \Sigma^* \times \Gamma^*$

$$\frac{\langle \overset{Q}{r}, \overset{\Gamma_\epsilon}{z} \rangle \in \delta(\overset{Q}{q}, \overset{\Sigma_\epsilon}{x}, \overset{\Gamma_\epsilon}{y})}{q, x\;\underset{\Sigma^*}{w}, y\;\underset{\Gamma^*}{v} \vdash r, w, z\;v}$$

Berekening

$\vdash$ relatie op $Q \times \Sigma^* \times \Gamma^*$

$$\frac{\langle r, z\rangle \in \delta(q, x, y)}{q, x\ w, y\ v \vdash r, w, z\ v}$$

$\langle\ Q, \Sigma, \Gamma, \delta, q_0, F\ \rangle$

Lemma

Een woord w wordt geaccepteerd door M precies als $q_0, w, \varepsilon \overset{*}{\vdash} r, \varepsilon, u$ met $r \in F$ voor een $u \in \Gamma^*$.

Algemenere PDA

Transitiefunctie veralgemeniseren naar

$$\delta : Q \times \Sigma_\varepsilon \times \Gamma_\varepsilon \to \mathbf{P}\left(Q \times \Gamma^*\right)$$

maakt PDA's niet wezenlijk sterker.

Andere Algemenere PDA

Eisen dat een berekening alleen accepterend is als de stack op ’t einde leeg is maakt PDA’s niet wezenlijk anders.

Context-vrij op PDA

Context-vrije talen worden herkent door PDA's

Context-vrij op PDA

Context-vrije talen worden herkent door PDA's

left-most $S \overset{*}{\Rightarrow} w\ u$ dan en slechts dan als $q_*, w, S \vdash q_*, \varepsilon, u$

Gezien

Context-Vrije Talen

PDA

CFG

Gezien

Context-Vrije Talen

PDA	CFG Greibach normaalvorm Chomsky normaalvorm

Gezien

Context-Vrije Talen

PDA
lange transities

CFG
Greibach normaalvorm
Chomsky normaalvorm