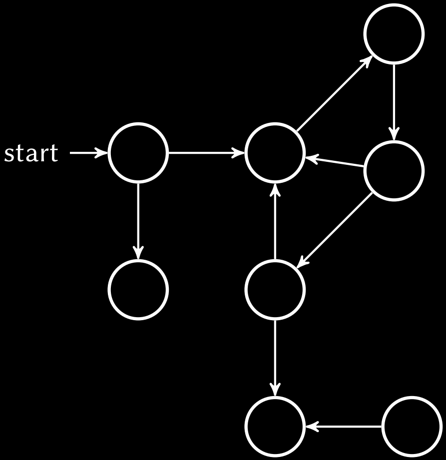
Logische Complexiteit

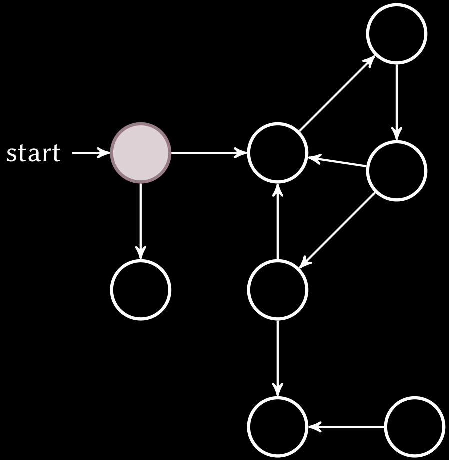
IX : Beslisbare Talen

Jeroen Goudsmit
Universiteit Utrecht
donderdag 8 maart 2012

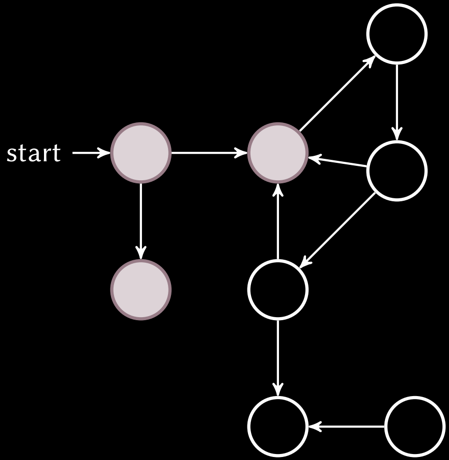
Knopen vinden



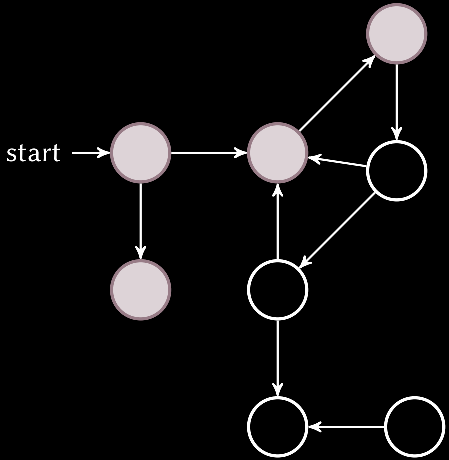
Knopen vinden



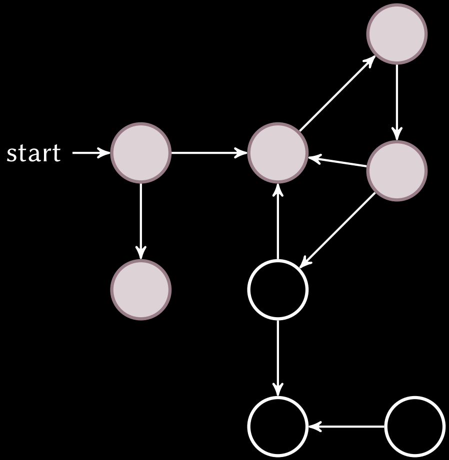
Knopen vinden



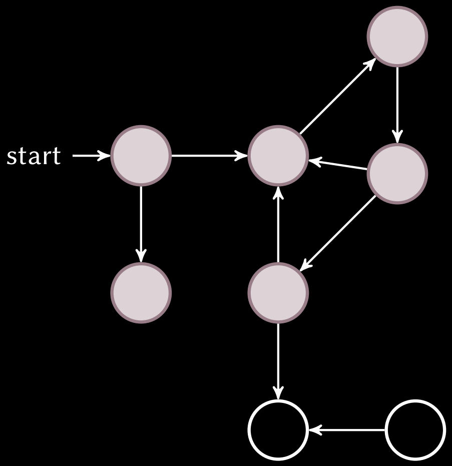
Knopen vinden



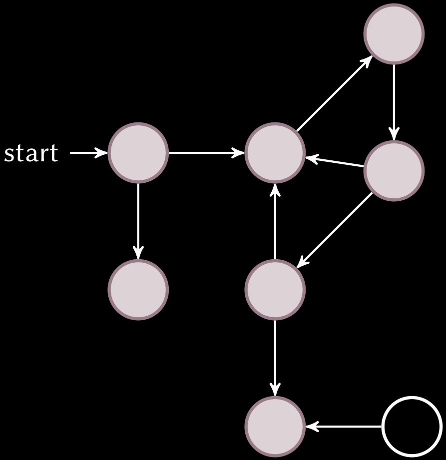
Knopen vinden



Knopen vinden



Knopen vinden



Knopen vinden: algoritme

M = "Op invoer $\langle V, E, v \rangle$:

1. Markeer knoop v.
2. Markeer elke knoop die bereikbaar is uit een gemarkeerde knoop.
3. Als er een nieuwe knoop gemarkeerd is, ga naar 2.
4. Is alles gemarkeerd, accepteer. Zoniet, verwerp."

Beslisbare Talen

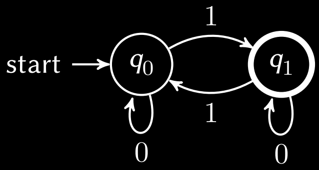
$\mathcal{L}$ is beslisbaar als er een beslisser M is met $\mathcal{L}(M) = \mathcal{L}$

Geaccepteerde Woorden

$$\mathrm{A}_{\textsc{dfa}} := \left\{ \langle M, w \rangle \;\middle|\; \begin{array}{l} M \text{ is een } \textsc{dfa} \text{ en} \\ \text{accepteert } w \end{array} \right\}$$

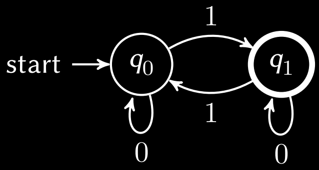
Voorbeeld

$\Big\langle$



$, 100101 \Big\rangle \overset{?}{\in} \mathrm{A}_{\mathsf{DFA}}$

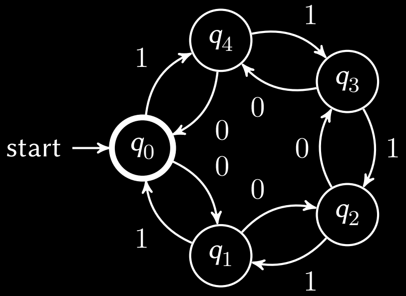
Voorbeeld

$\Big\langle$ 

$, 100101 \Big\rangle \in A_{\mathsf{DFA}}$

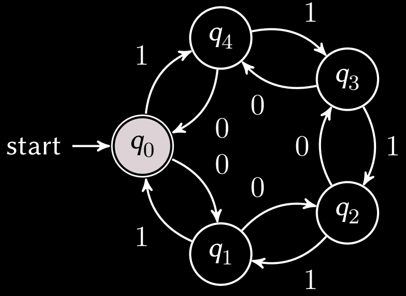
Voorbeeld Generations

$\Big\langle$



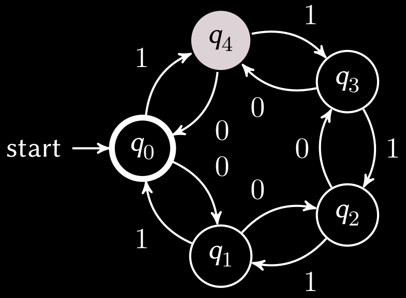
$, \ 10100100010000 \Big\rangle \stackrel{?}{\in} A_{\mathrm{DFA}}$

Voorbeeld Generations

$\Big\langle$ 

$, \ 10100100010000 \Big\rangle \overset{?}{\in} \mathrm{A}_{\mathrm{DFA}}$

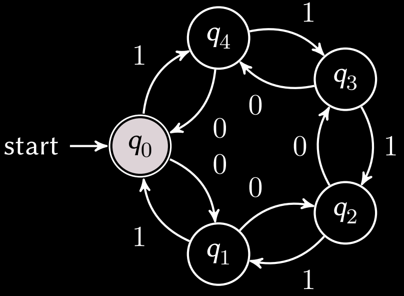
Voorbeeld Generations

$\Big\langle$ 

$, \; 10100100010000 \Big\rangle \overset{?}{\in} \mathrm{A}_{\textsc{dfa}}$

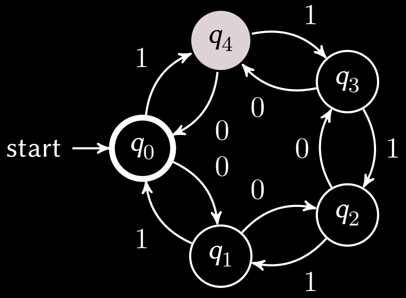
Voorbeeld Generations

$\Bigg\langle$



$, \ 10100100010000 \ \Bigg\rangle \overset{?}{\in} \mathrm{A_{DFA}}$

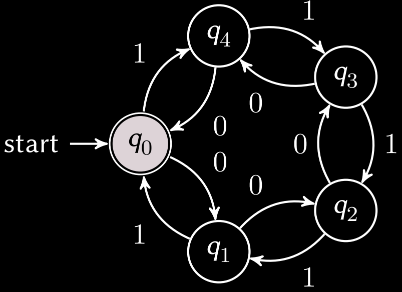
Voorbeeld Generations



$$\left\langle \quad , \ 10100100010000 \right\rangle \overset{?}{\in} \mathrm{A}_{\mathrm{DFA}}$$

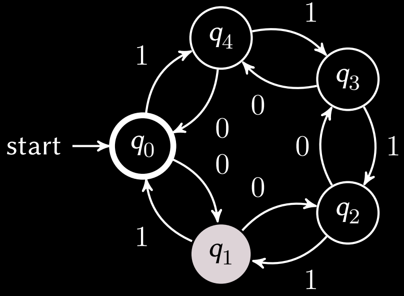
Voorbeeld Generations

$\Big\langle$



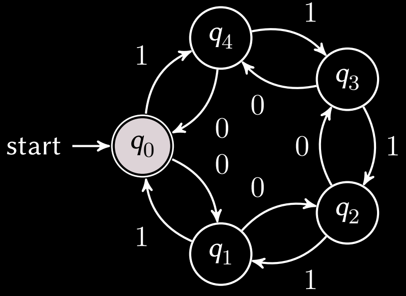
$, \; 10100100010000 \; \Big\rangle \stackrel{?}{\in} \mathrm{A}_{\mathrm{DFA}}$

Voorbeeld Generations



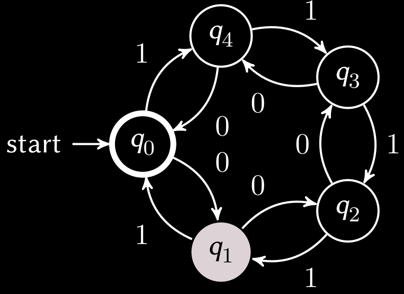
$$\left\langle \text{DFA} \;,\; 10100100010000 \right\rangle \overset{?}{\in} A_{\mathrm{DFA}}$$

Voorbeeld Generations

$\Big\langle$ 

$, \ 10100100010000 \Big\rangle \overset{?}{\in} \mathrm{A}_{\text{DFA}}$

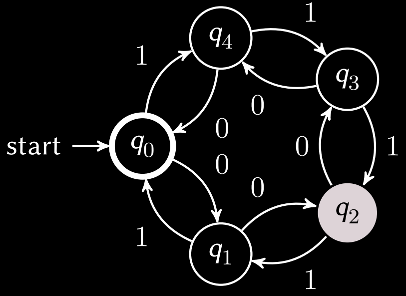
Voorbeeld Generations

$\Big\langle$ 

$, \ 10100100010000 \ \Big\rangle \stackrel{?}{\in} A_{\text{DFA}}$

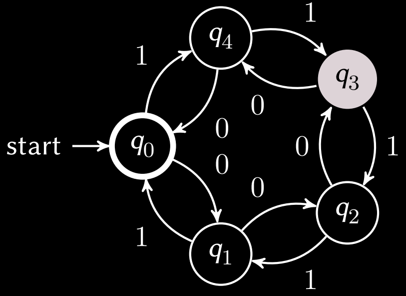
Voorbeeld Generations

$\Big\langle$



$, \ 10100100010000 \Big\rangle \overset{?}{\in} A_{\text{DFA}}$

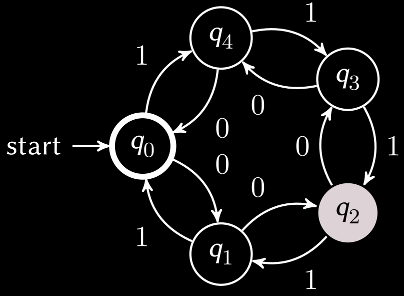
Voorbeeld Generations

$\Big\langle$ 

$, \; 10100100010000 \Big\rangle \overset{?}{\in} A_{DFA}$

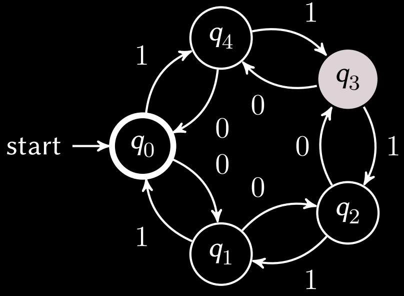
Voorbeeld Generations

$\Big\langle$



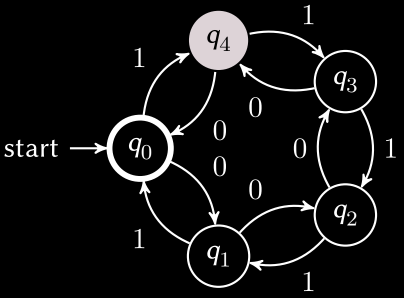
$, \ 10100100010000 \ \Big\rangle \overset{?}{\in} \mathrm{A}_{\mathsf{DFA}}$

Voorbeeld Generations

$\Big\langle$ 

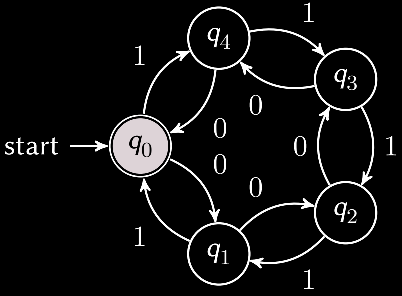
$, \; 10100100010000 \; \Big\rangle \overset{?}{\in} A_{\text{DFA}}$

Voorbeeld Generations

$\Big\langle$ 

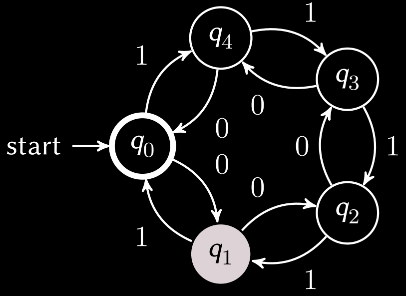
, 10100100010000 $\Big\rangle \stackrel{?}{\in} A_{\text{DFA}}$

Voorbeeld Generations

$\Big\langle$ 

, $10100100010000 \Big\rangle \stackrel{?}{\in} \mathrm{A}_{\mathrm{DFA}}$

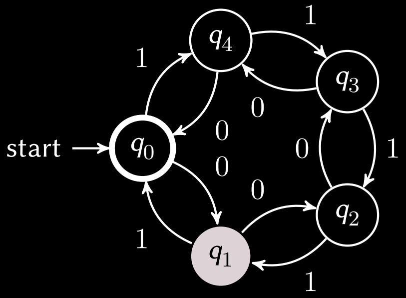
Voorbeeld Generations



$$\Big\langle \;\text{(DFA)}\; ,\; 10100100010000 \Big\rangle \overset{?}{\in} A_{\text{DFA}}$$

Voorbeeld Generations

$\langle$



$, \; 10100100010000 \rangle \notin A_{\mathrm{DFA}}$

Geaccepteerde Woorden

M = "Op invoer $\langle D, w \rangle$ met D een DFA:

1. Simuleer D op w.
2. Als D accepteert, accepteer. Anders, verwerp."

Geaccepteerde Woorden

beslist A_{DFA}

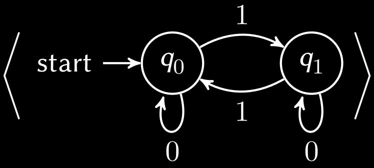
M = "Op invoer $\langle D, w \rangle$ met D een DFA:

1. Simuleer D op w.
2. Als D accepteert, accepteer. Anders, verwerp."

Lege Machines

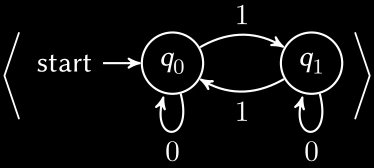
$$\mathrm{E}_{\textsc{dfa}} := \left\{ \langle M \rangle \;\middle|\; \begin{array}{l} M \text{ is een } \textsc{dfa} \text{ en} \\ \text{accepteert niks} \end{array} \right\}$$

Voorbeeld



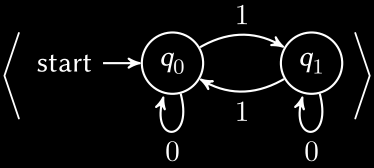
$\overset{?}{\in} E_{\text{DFA}}$

Voorbeeld

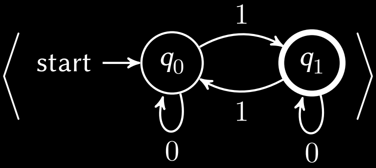


$\in \mathrm{E}_{\mathrm{DFA}}$

Voorbeeld

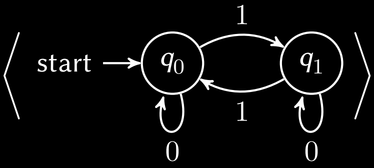


$\in E_{\text{DFA}}$

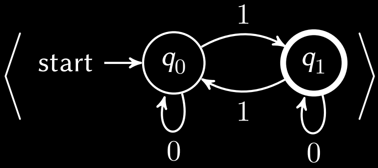


$\stackrel{?}{\in} E_{\text{DFA}}$

Voorbeeld

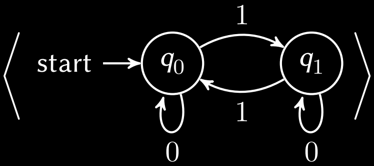


$\in \mathrm{E}_{\text{DFA}}$

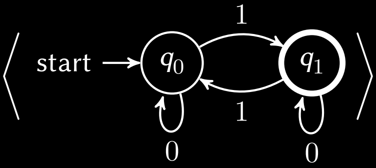


$\notin \mathrm{E}_{\text{DFA}}$

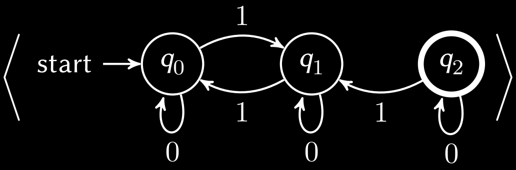
Voorbeeld



$\in \mathrm{E_{DFA}}$

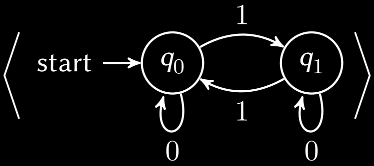


$\notin \mathrm{E_{DFA}}$

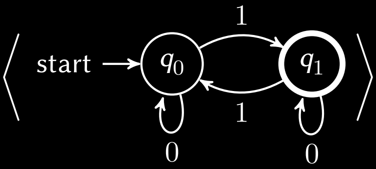


$\overset{?}{\in} \mathrm{E_{DFA}}$

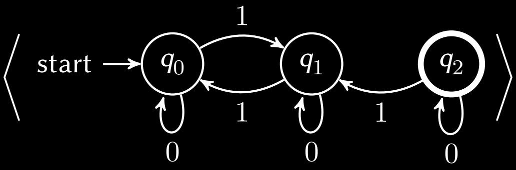
Voorbeeld



$\in \mathrm{E}_{\mathrm{DFA}}$

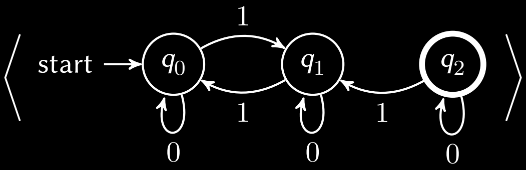


$\notin \mathrm{E}_{\mathrm{DFA}}$



$\in \mathrm{E}_{\mathrm{DFA}}$

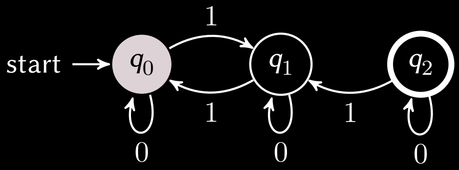
Voorbeeld First Contact



$\overset{?}{\in} \mathrm{E}_{\mathrm{DFA}}$

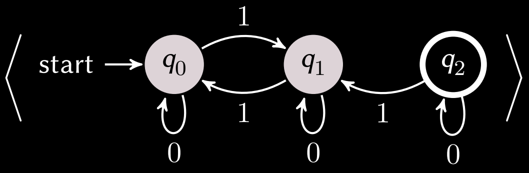
Voorbeeld First Contact

$\Big\langle$



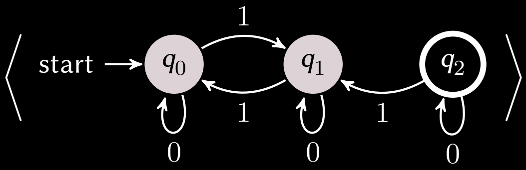
$\Big\rangle \overset{?}{\in} \mathrm{E}_{\mathsf{DFA}}$

Voorbeeld First Contact



$\stackrel{?}{\in} \mathrm{E}_{\text{DFA}}$

Voorbeeld First Contact



$\in \mathrm{E}_{\mathrm{DFA}}$

Lege Machines

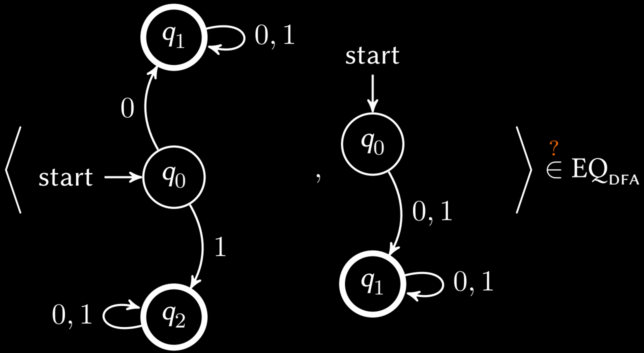
M = "Op invoer van een CFG $\langle Q, \Sigma, \delta, q_0, F \rangle$:

1. Markeer q_0.
2. Voor elke gemarkeerde toestand, markeer alle via δ bereikbare toestanden.
3. Als er een nieuwe toestand gemarkeerd is, ga naar 2.
4. Test of er een toestand uit F gemarkeerd is. Zo ja, accepteer. Zo nee, verwerp."

Equivalente Machines

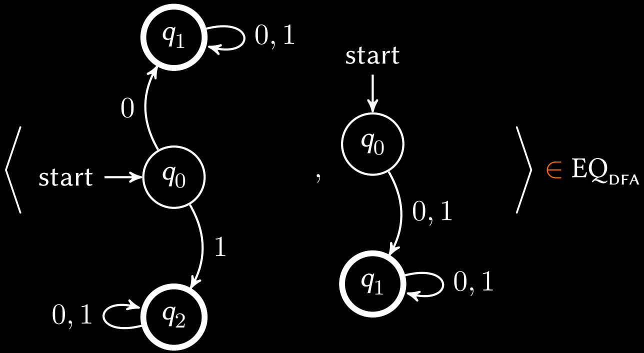
$$\mathrm{EQ}_{\textsc{dfa}} := \left\{ \langle M, N \rangle \;\middle|\; \begin{array}{l} M \text{ en } N \text{ zijn } \textsc{dfa}\text{'s en} \\ \text{accepteren hetzelfde} \end{array} \right\}$$

Voorbeeld



start
q_0
0
q_1
0, 1
1
q_2
0, 1
,
start
q_0
0, 1
q_1
0, 1
$\stackrel{?}{\in} \mathrm{EQ}_{\mathrm{DFA}}$

Voorbeeld



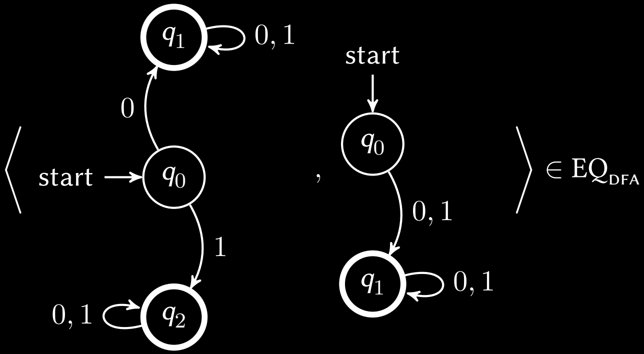
Symmetrisch Verschil

$X = Y$ precies wanneer $X - Y$ en $Y - X$ beiden leeg

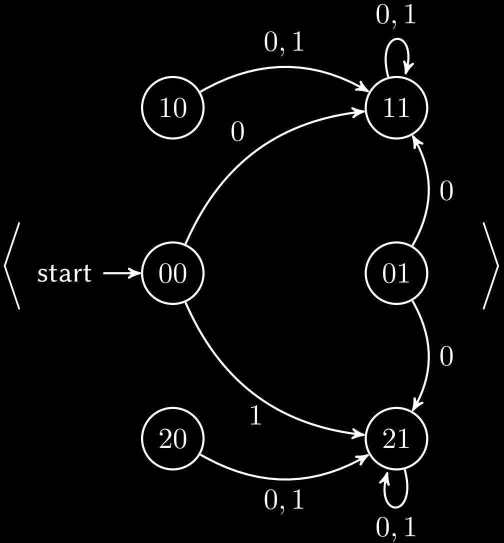
Symmetrisch Verschil

$X = Y$ precies wanneer $X - Y$ en $Y - X$ beiden leeg
precies wanneer $\left(X - Y\right) \cup \left(Y - X\right)$ leeg

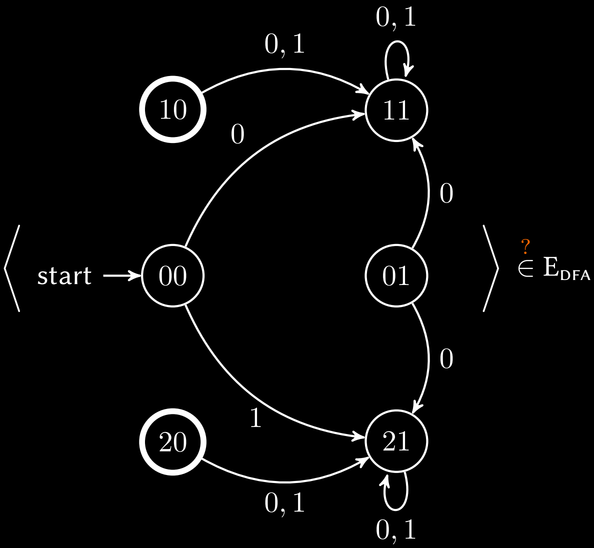
Voorbeeld Insurrection



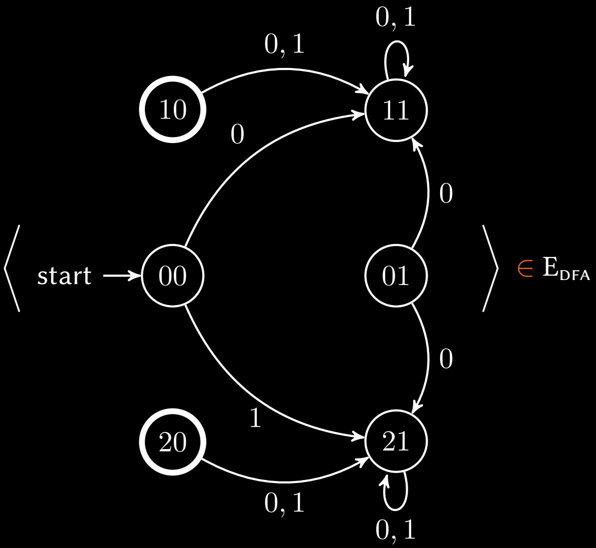
Voorbeeld Insurrection



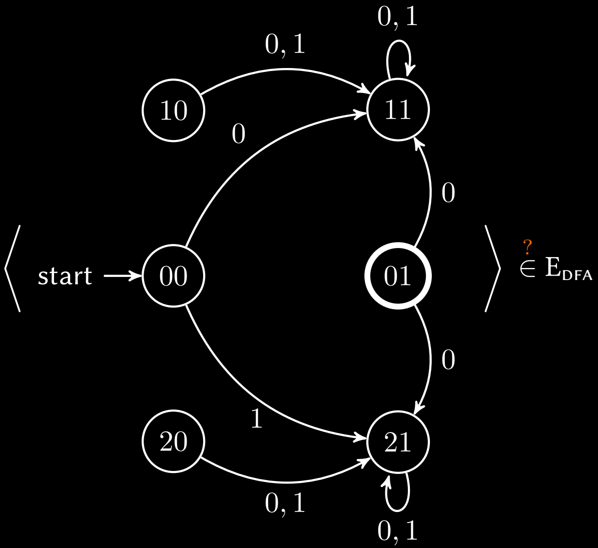
Voorbeeld Insurrection



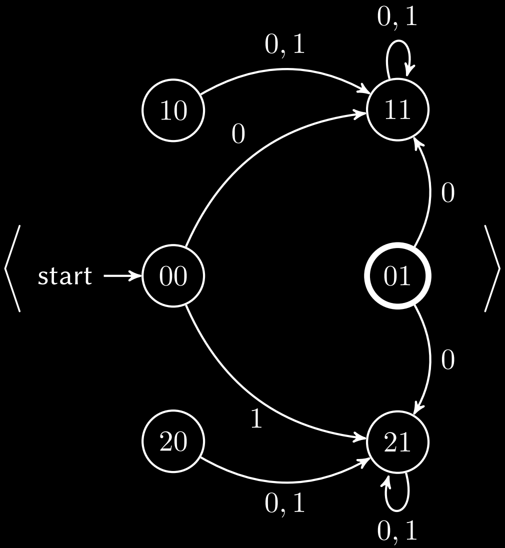
Voorbeeld Insurrection



Voorbeeld Insurrection



Voorbeeld Insurrection



$\in \mathrm{E}_{\mathsf{DFA}}$

Equivalente Machines

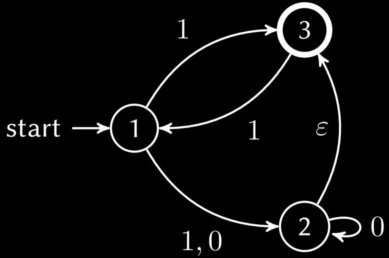
M = "Op invoer $\langle P, Q \rangle$ met P en Q DFA's:

1. Construeer S voor symmetrisch verschil P en Q.
2. Test of $S \in \mathrm{E}_{\mathrm{DFA}}$. Zo ja, accepteer. Zo nee, verwerp."

Accepterende Woorden (NFA)

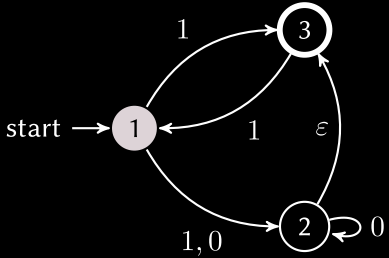
$$\mathrm{A}_{\mathrm{NFA}} := \left\{ \langle M, w \rangle \;\middle|\; \begin{array}{l} M \text{ is een NFA en} \\ \text{accepteert } w \end{array} \right\}$$

Voorbeeld

$\Big\langle$ 

$, 0011 \Big\rangle \stackrel{?}{\in} A_{\mathsf{NFA}}$

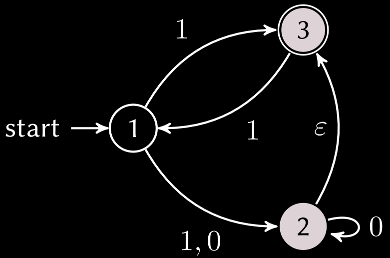
Voorbeeld

$\Big\langle$ 

$, 0011 \Big\rangle \stackrel{?}{\in} A_{\mathsf{NFA}}$

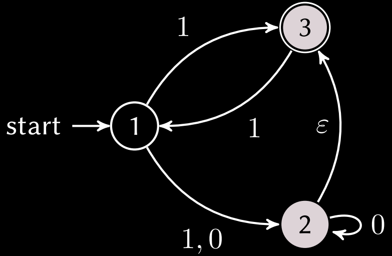
Voorbeeld

$\Big\langle$



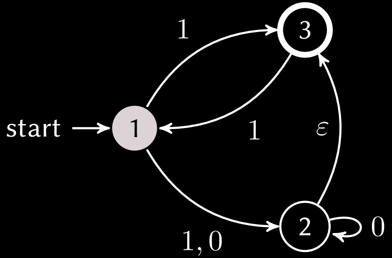
$, 0011 \Big\rangle \overset{?}{\in} A_{\text{NFA}}$

Voorbeeld



$$\left\langle \quad , 0011 \right\rangle \overset{?}{\in} A_{\mathsf{NFA}}$$

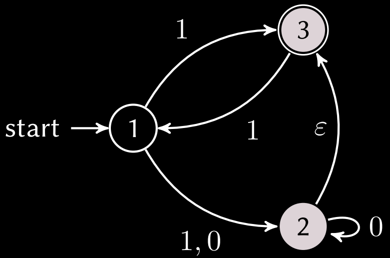
Voorbeeld

$\Big\langle$ 

$, 0011 \Big\rangle \overset{?}{\in} A_{\mathsf{NFA}}$

Voorbeeld

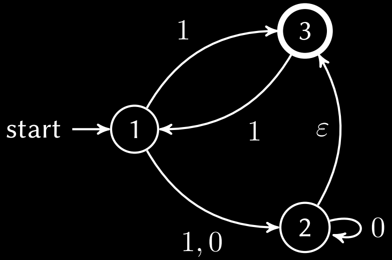
$\Big\langle$



$, 0011 \Big\rangle \overset{?}{\in} A_{\mathsf{NFA}}$

Voorbeeld

$\Big\langle$



$, 0011 \Big\rangle \in A_{\mathsf{NFA}}$

Geaccepteerde Woorden (CFG)

$$\mathrm{A}_{\mathrm{CFG}} := \left\{ \langle G, w \rangle \;\middle|\; \begin{array}{l} G \text{ is een CFG en} \\ \text{genereert } w \end{array} \right\}$$

Voorbeeld

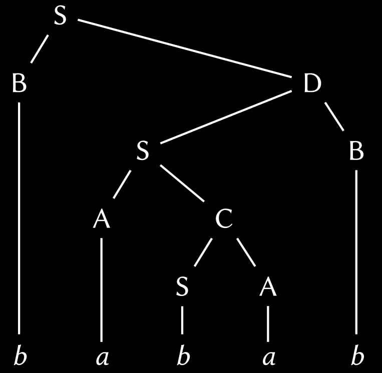
$$\left\langle \begin{array}{l} S \to a\,S\ \mathrm{a} \\ S \to b\,S\ \mathrm{b} \\ S \to a \\ S \to b \\ \end{array}, \quad b \quad a \quad b \quad a \quad b \right\rangle \overset{?}{\in} \mathrm{A_{CFG}}$$

Voorbeeld

$$\left\langle \begin{array}{l} S \to a\ C \\ S \to b\ D \\ S \to a \\ S \to b \\ A \to a \\ B \to b \\ C \to S\ A \\ D \to S\ B \end{array} ,\quad b \quad a \quad b \quad a \quad b \right\rangle \stackrel{?}{\in} A_{\text{CFG}}$$

Voorbeeld

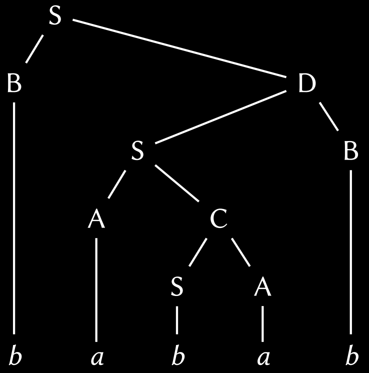
$$
\left\langle\ \begin{array}{l}
S \to a\ C \\
S \to b\ D \\
S \to a \\
S \to b \\
A \to a \\
B \to b \\
C \to S\ A \\
D \to S\ B
\end{array}\ ,\right.
$$



$$\left.\right\rangle \overset{?}{\in} A_{\mathrm{CFG}}$$

Voorbeeld

$$\left\langle \begin{array}{l} S \rightarrow a\ C \\ S \rightarrow b\ D \\ S \rightarrow a \\ S \rightarrow b \\ A \rightarrow a \\ B \rightarrow b \\ C \rightarrow S\ A \\ D \rightarrow S\ B \end{array} , \right.$$



$$\rangle \in A_{\mathrm{CFG}}$$

Geaccepteerde Woorden (CFG)

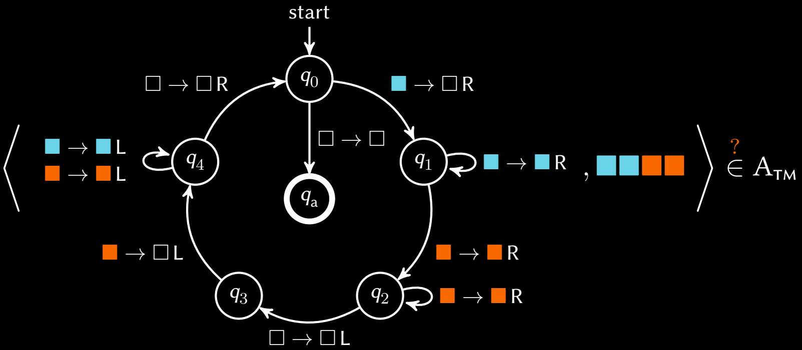
M = "Op invoer $\langle\, G, w \,\rangle$ met G een CFG en w een woord:

1. Construeer een CFG H in Chomsky-normaalvorm voor G.
2. Meet w, noem haar lengte n.
3. Als $n = 0$, maak alle afleidingen in H van lengte 1.
3. Als $n \geq 1$, maak alle afleidingen in H van lengte $2n - 1$.
4. Test of een van de afleidingen w geeft. Zo ja, accepteer. Zo nee, verwerp."

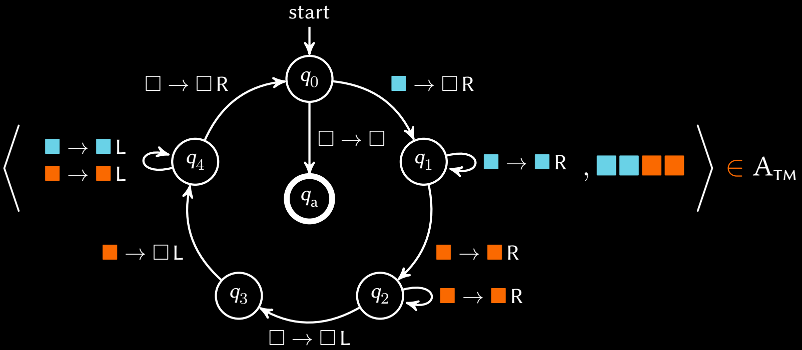
Geaccepteerde Woorden (TM)

$$\mathrm{A}_{\mathrm{TM}} := \left\{ \langle M, w \rangle \;\middle|\; \begin{array}{l} M \text{ is een TM en} \\ \text{accepteert } w \end{array} \right\}$$

Voorbeeld



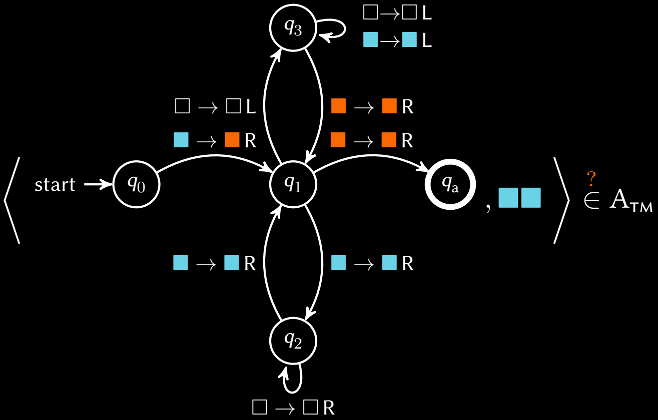
Voorbeeld



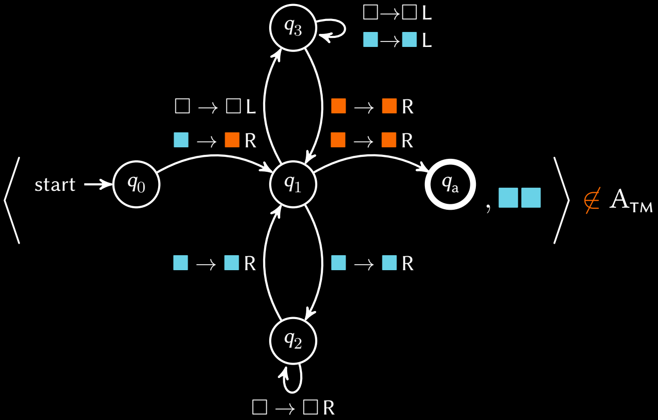
Geaccepteerde Woorden (TM)

M = "Op invoer van $\langle N, w \rangle$ met N een TM:
1. Simuleer N op w."

Voorbeeld



Voorbeeld



Geaccepteerde Woorden (TM)

A_{TM} is niet beslisbaar

(bewijs volgt later)

Lege Machines (CFG)

$$\mathrm{E}_{\textsc{cfg}} := \left\{ \langle G \rangle \;\middle|\; \begin{array}{l} G \text{ is een CFG en} \\ \text{genereert niks} \end{array} \right\}$$

Voorbeeld

$$\left\langle \begin{array}{l} S \rightarrow S\ a \\ S \rightarrow B\ a \\ A \rightarrow b\ A \\ B \rightarrow a \\ B \rightarrow a\ B \end{array} \right\rangle \stackrel{?}{\in} E_{\text{CFG}}$$

Voorbeeld

$$\left\langle \begin{array}{l} \mathrm{S} \rightarrow \mathrm{S}\ a \\ \mathrm{S} \rightarrow \mathrm{B}\ a \\ \mathrm{A} \rightarrow b\ \mathrm{A} \\ \mathrm{B} \rightarrow a \\ \mathrm{B} \rightarrow a\ \mathrm{B} \end{array} \right\rangle \overset{?}{\in} \mathrm{E}_{\mathrm{CFG}}$$

Voorbeeld

$$\left\langle \begin{array}{l} \mathrm{S} \rightarrow \mathrm{S}\ a \\ \mathrm{S} \rightarrow \mathrm{B}\ a \\ \mathrm{A} \rightarrow b\ \mathrm{A} \\ \mathrm{B} \rightarrow a \\ \mathrm{B} \rightarrow a\ \mathrm{B} \end{array} \right\rangle \stackrel{?}{\in} \mathrm{E}_{\mathrm{CFG}}$$

Voorbeeld

$$\left\langle \begin{array}{l} S \rightarrow S\ a \\ S \rightarrow B\ a \\ A \rightarrow b\ A \\ B \rightarrow a \\ B \rightarrow a\ B \end{array} \right\rangle \in E_{\text{CFG}}$$

Lege Machines (CFG)

M = "Op invoer van een DFA $\langle\, V, \Sigma, R, S \,\rangle$:

1. In elke regel, markeer elk symbool uit Σ.
2. Per regel $A \to w$: markeer A overal, als ieder symbool uit w gemarkeerd is.
3. Als er iets nieuws gemarkeerd is, ga naar 2.
4. Test of S gemarkeerd is. Zo ja, verwerp. Zo nee, accepteer."

Lege Machines (TM)

$$\mathrm{E}_{\text{TM}} := \left\{ \langle M \rangle \;\middle|\; \begin{array}{l} M \text{ is een TM en} \\ \text{accepteert niks} \end{array} \right\}$$

Lege Machines (TM)

Lege Machines (TM)

Per TM N en woord v maak $L_{N,v}$.

$L_{N,v}$ = "Op invoer w:

1. Als $v \neq w$, verwerp
1. Anders, simuleer N op v.
2. Als N accepteert, accepteer."

Lege Machines (TM)

Per TM N en woord v maak $L_{N,v}$.

$L_{N,v}$ = "Op invoer w:

1. Als $v \neq w$, verwerp
1. Anders, simuleer N op v.
2. Als N accepteert, accepteer."

Als N het woord v accepteert, dan accepteert $L_{N,v}$ ook v.

Lege Machines (TM)

Per TM N en woord v maak $L_{N,v}$.

$L_{N,v}$ = "Op invoer w:

1. Als $v \neq w$, verwerp
1. Anders, simuleer N op v.
2. Als N accepteert, accepteer."

Als N het woord v accepteert, dan accepteert $L_{N,v}$ ook v.

Als $L_{N,v}$ iets accepteert, dan accepteert N het woord v.

Lege Machines (TM)

Per TM N en woord v maak $L_{N,v}$.

$L_{N,v}$ = "Op invoer w:

1. Als $v \neq w$, verwerp
1. Anders, simuleer N op v.
2. Als N accepteert, accep-teer."

Als N het woord v accepteert, dan accepteert $L_{N,v}$ ook v.

Als $L_{N,v}$ iets accepteert, dan accepteert N het woord v.

$L_{N,v} \notin \mathrm{E}_{\mathrm{TM}}$ precies als $\langle N, v \rangle \in \mathrm{A}_{\mathrm{TM}}$.

Lege Machines (TM)

Per TM N en woord v maak $L_{N,v}$.

$L_{N,v}$ = "Op invoer w:

1. Als $v \neq w$, verwerp
1. Anders, simuleer N op v.
2. Als N accepteert, accepteer."

Als N het woord v accepteert, dan accepteert $L_{N,v}$ ook v.

Als $L_{N,v}$ iets accepteert, dan accepteert N het woord v.

$L_{N,v} \notin \mathrm{E}_{\mathrm{TM}}$ precies als $\langle N, v \rangle \in \mathrm{A}_{\mathrm{TM}}$.

Als E_{TM} beslisbaar zou zijn, dan was A_{TM} 't ook!

Lege Machines

E_{TM} is niet beslisbaar

(bewijs rond wanneer A_{TM} niet beslisbaar)

Equivalente Grammatica's

Equivalente Grammatica's

EQ_{CFG} is niet beslisbaar

(bewijs volgt later)

Equivalente Grammatica's

EQ_{CFG} is niet beslisbaar

(bewijs volgt later)

(minstens zo moeilijk als A_{TM})

Equivalente Machines (TM)

Equivalente Machines (TM)

Als EQ_{TM} beslisbaar zou zijn, dan was E_{TM} 't ook!

Talen

	A	E	EQ
DFA			
CFG			
TM			

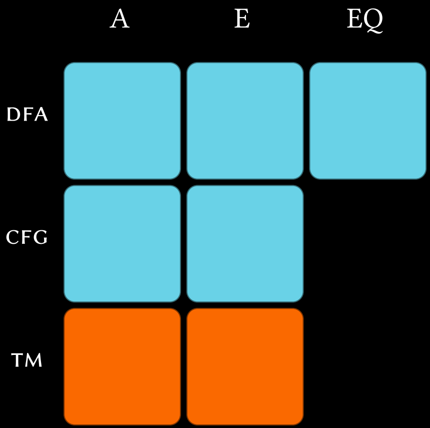
Talen

	A	E	EQ
DFA			
CFG			
TM			

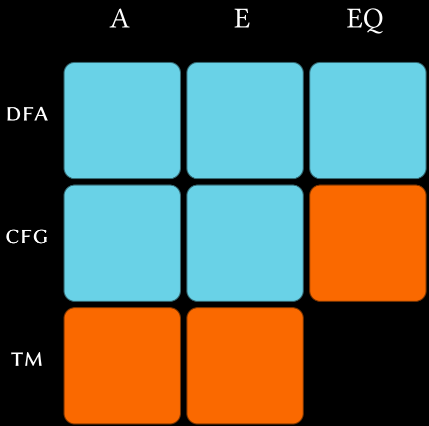
Talen

	A	E	EQ
DFA			
CFG			
TM			

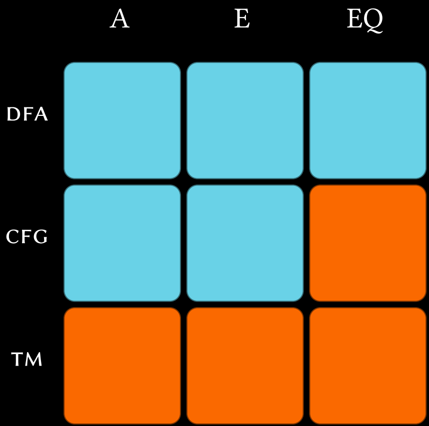
Talen



Talen



Talen



Gezien

Er zijn interessante beslisbare talen

Gezien

Er zijn interessante beslisbare talen

Sommige zijn “gemakkelijker” dan anderen.