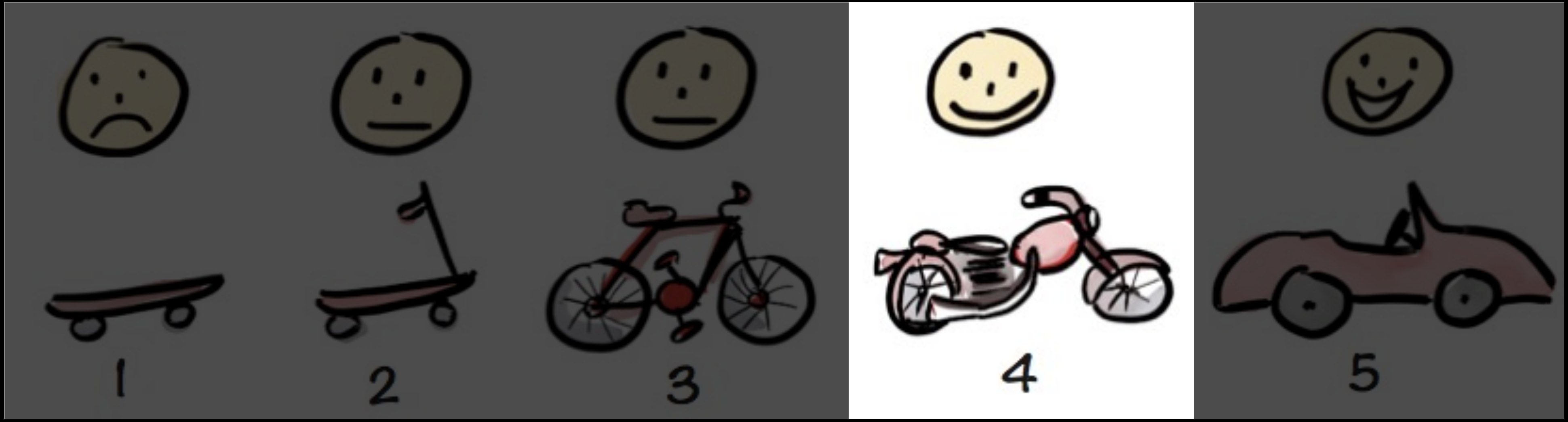
Errungenschaften der letzten Vorlesung



Die Syntax einer formalen Sprache wird mit **formalen Grammatiken** spezifiziert. **Rechts- und linkslineare Grammatiken** lassen sich mit endlichen Automaten verarbeiten. Für andere Grammatiken muss der **Syntaxbaum** noch manuell ermittelt werden. Alle Token, für die ein **endlicher Akzeptor** existiert, können automatisch zu einem **Tokenizer / Lexer** kombiniert werden. Zu jedem NEA kann mithilfe der **Potenzmengenkonstruktion** ein äquivalenter DEA berechnet werden.

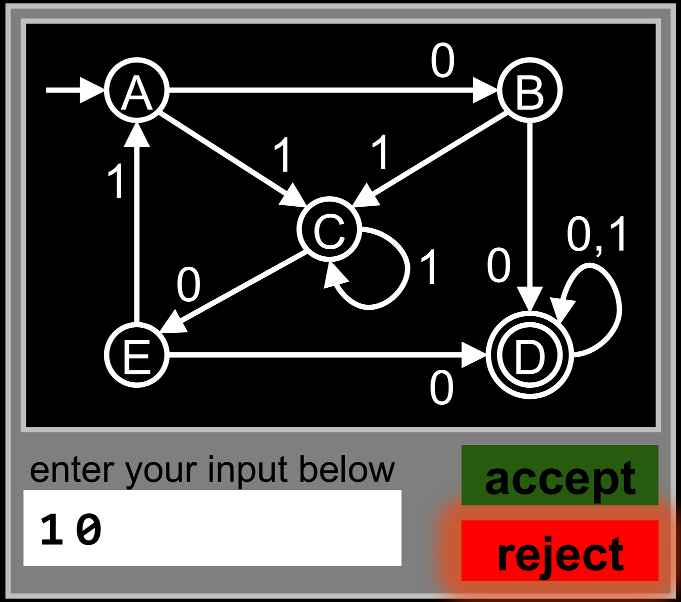
Offene Fragen

1. ~~Nicht jedes Token darf an jeder Stelle stehen. Wie lässt sich das regulieren?~~
2. ~~Sind die von endlichen Automaten akzeptierten Sprachen identisch zu den von Grammatiken erzeugten Sprachen?~~
3. ~~Lassen sich alle NEAs in DEAs umwandeln?~~
4. ~~Sind rechtslineare und linkslineare Grammatiken verschieden?~~
5. Lassen sich endliche Automaten minimieren?
6. Sind unbeschränkte Grammatiken mächtiger als die linearen?
7. Kann man die Erstellung eines endlichen Automaten automatisieren?

Was ist mit “minimieren” gemeint?

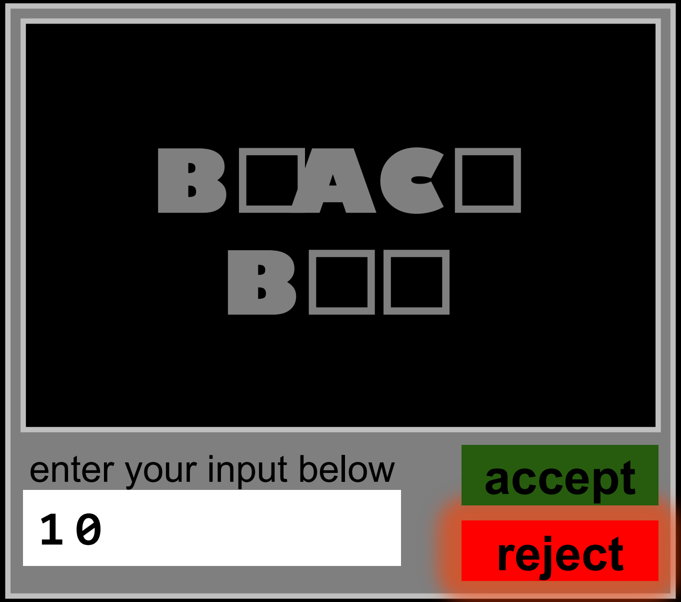
Anforderung des Kunden:
“Akzeptiere alle Zeichenketten über dem Alphabet $\{0,1\}$ mit dem **Teilwort 00**.”

Unser Automatenmodell:



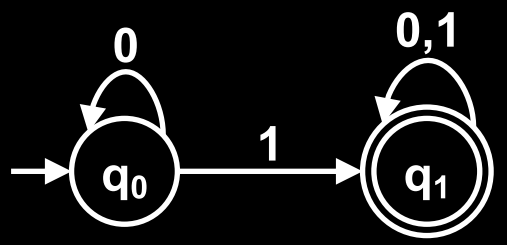
Den Kunden interessiert das zugrundeliegende Automatenmodell nicht. Wenn es für die gegebene Sprache einen weiteren Automaten mit **weniger Zuständen** gibt, kann dieser einfach ausgetauscht werden. Eine geringere Zustandszahl **senkt den Speicherbedarf** und ist deshalb in jedem Fall zu favorisieren.

Sicht des Kunden:



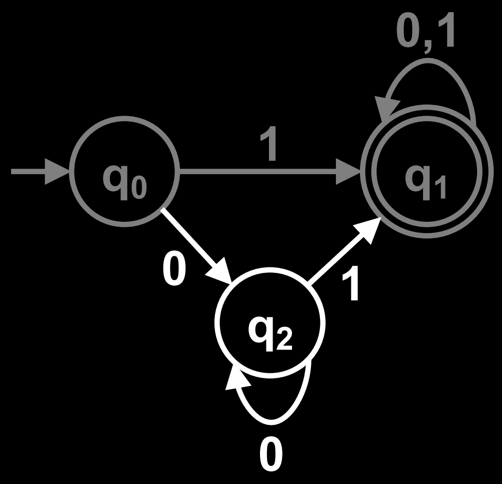
Lassen sich endliche Automaten minimieren?

Es dürfte schwer fallen, intuitiv ein Konzept zur Minimierung eines endlichen Akzeptors zu entwickeln. Viel einfacher ist es hingegen, einen Akzeptor zu “maximieren”. Lasst uns versuchen durch diese Problemumkehr Erkenntnisse zu gewinnen, die uns bei der Minimierung dienlich sind. Beginnen wir mit diesem Akzeptor:



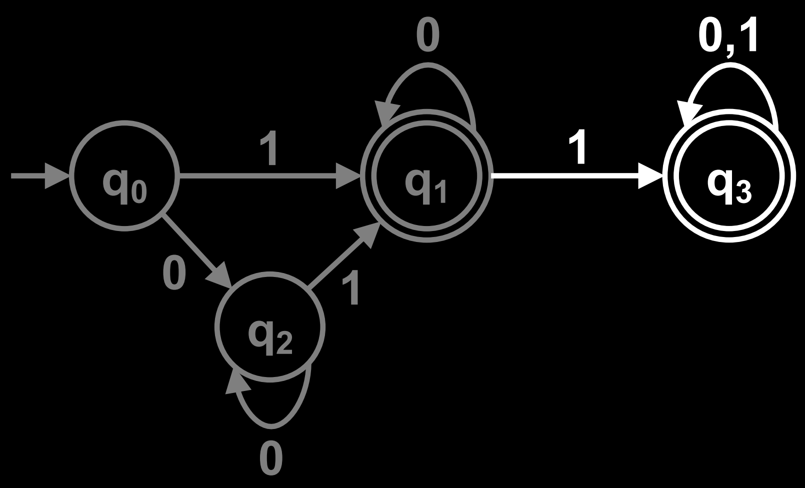
Lassen sich endliche Automaten minimieren?

Bei Eingabe von 0 im Zustand q_0 verblieb der Automat in q_0. Diese Schleife könnten wir "ausrollen", indem wir in einen neuen Zustand q_2 überführen, welcher dieselben Transitionen wie q_0 besitzt. Die Sprache wird dadurch nicht verändert, aber der Automat hat nun einen Zustand mehr.



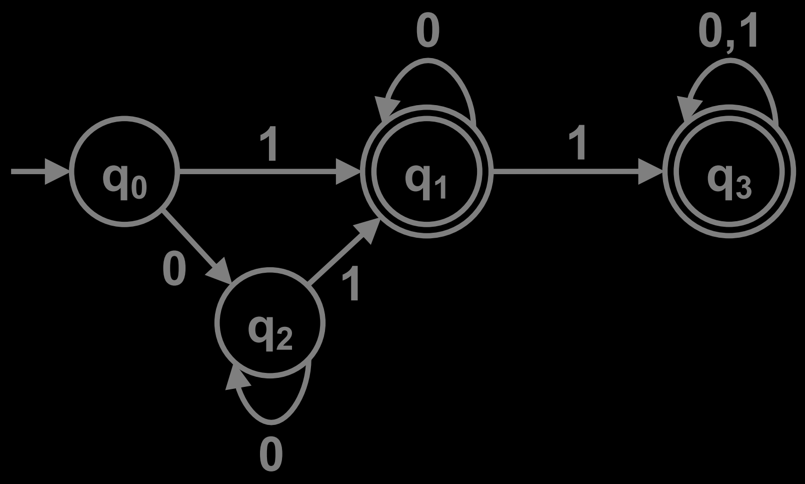
Lassen sich endliche Automaten minimieren?

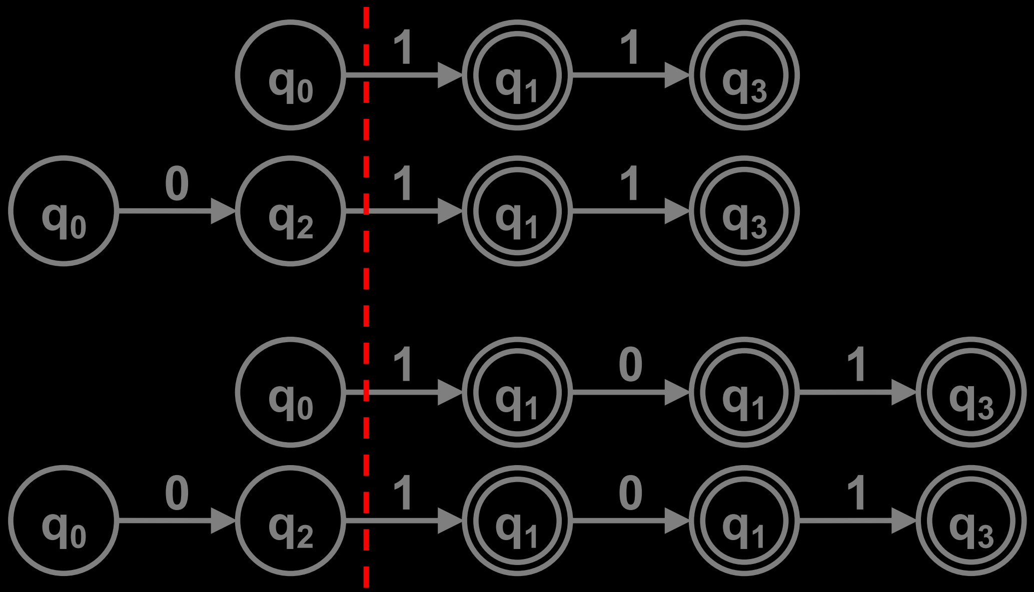
Dieselbe Maßnahme wie zuvor bei q_0 können wir natürlich auch bei q_1 ergreifen. Auch in dieser Situation ändert sich die erkannte Sprache nicht.



Lassen sich endliche Automaten minimieren?

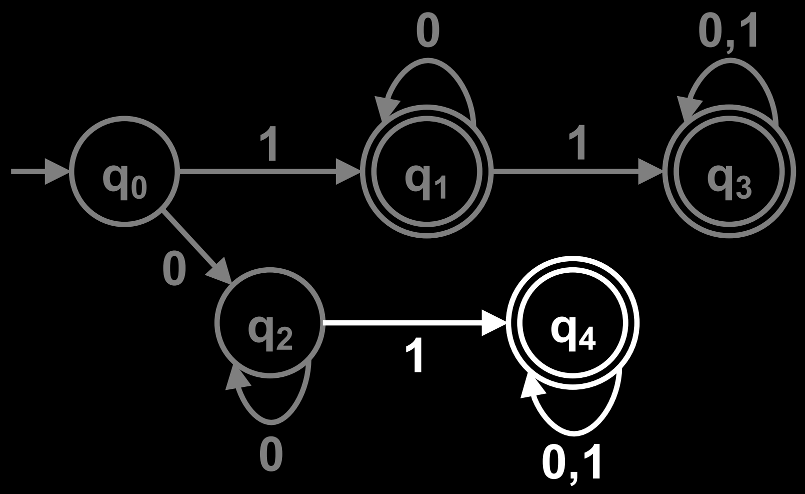
Aufgrund unserer Konstruktion wissen wir, dass die Zustände q_0 und q_2 sowie q_1 und q_3 äquivalent sind. Wenn wir uns nun verschiedene Läufe durch den Akzeptor ansehen, fällt auf, dass die Zustandsabfolgen hinter den beiden äquivalenten Zuständen identisch sind.





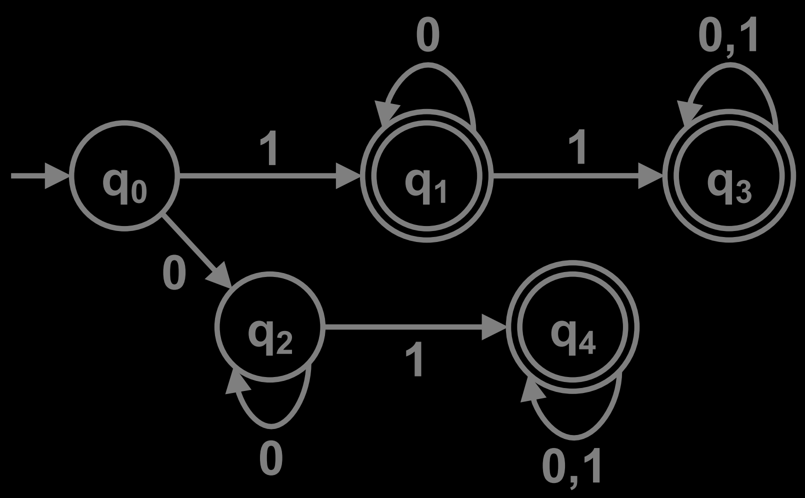
Lassen sich endliche Automaten minimieren?

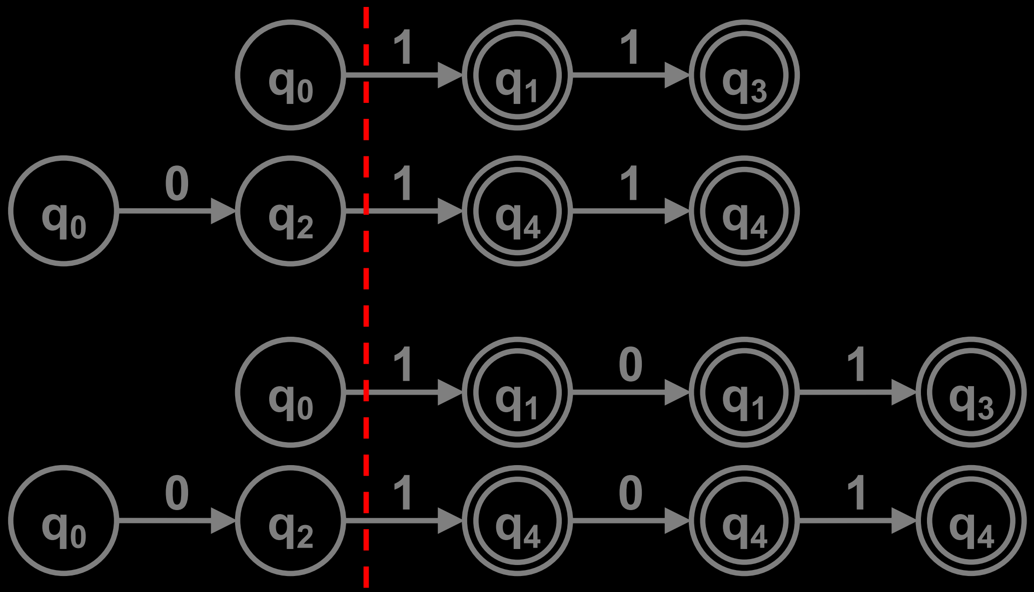
Wir können jedoch erzwingen, dass die Zustandsabfolge unterschiedlich ist, indem wir aus Zustand q_2 in einen neuen Zustand q_4 überführen, der q_1 entsprichtt.



Lassen sich endliche Automaten minimieren?

Nun ist die Zustandsfolge hinter den beiden äquivalenten Zuständen zwar verschieden, ihr Akzeptanzverhalten bleibt davon jedoch unberührt. Ausgehend von den beiden Zuständen müssen also die gleichen Sprachen erkannt werden.





$\{\varepsilon\} \circ L(M_{q_0})$

$\{0\} \circ L(M_{q_2})$

$\{\varepsilon\} \circ L(M_{q_0})$

$\{0\} \circ L(M_{q_2})$

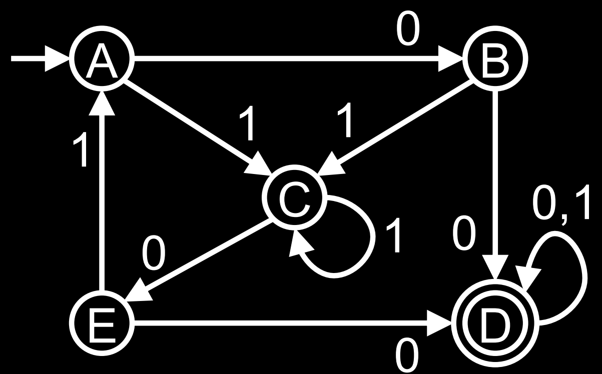
Erste Erkenntnisse

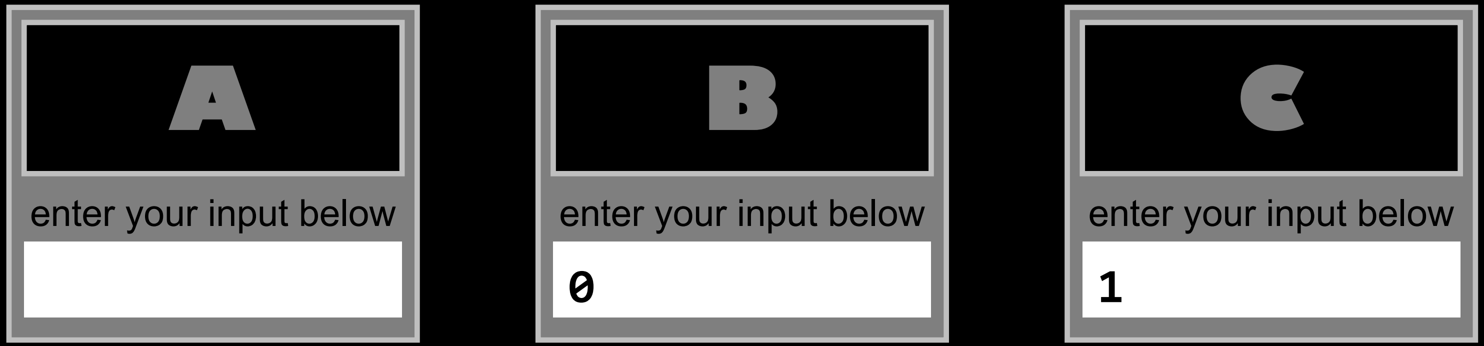
Ein vollständiger DFA M heißt minimal, wenn kein anderer vollständiger DFA für $L(M)$ weniger Zustände hat als M.

Sei $M := (\Sigma, Q, \delta, q_0, F)$ ein vollständiger DFA und M_p derselbe DFA, aber mit Startzustand p.

Zwei Zustände $p, q \in Q$ sind nicht unterscheidbar, wenn $L(M_p) = L(M_q)$. Wir notieren dies als $p \equiv_M q$.

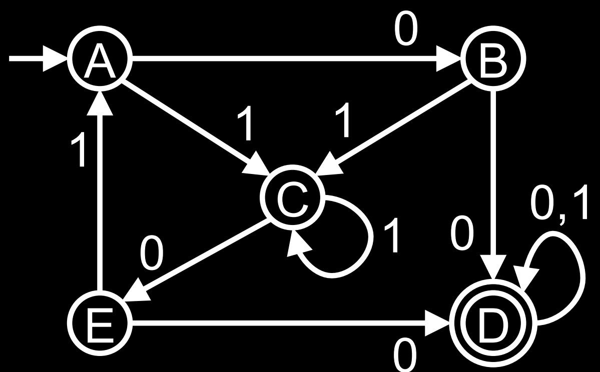
Welche Zustände sind unterscheidbar?

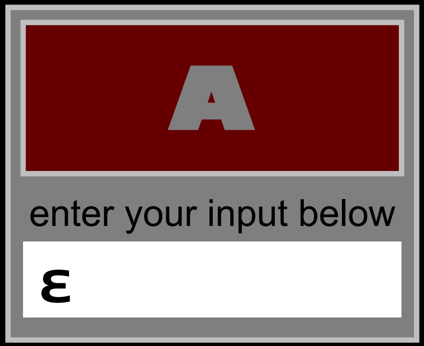


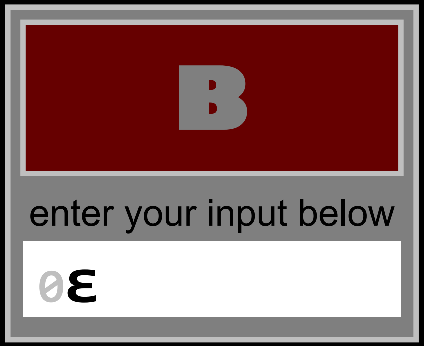


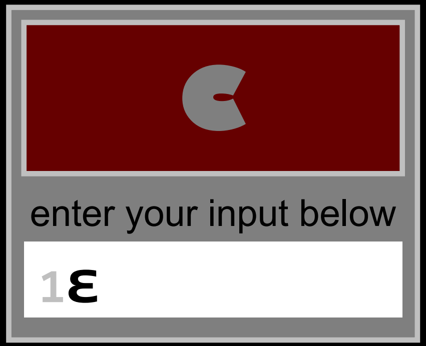
Mit den dargestellten Eingaben versetzen wir fünf Instanzen des Automaten in alle möglichen Zustände und betrachten sie von da an als Blackbox, um ihr Akzeptanzverhalten zu analysieren.

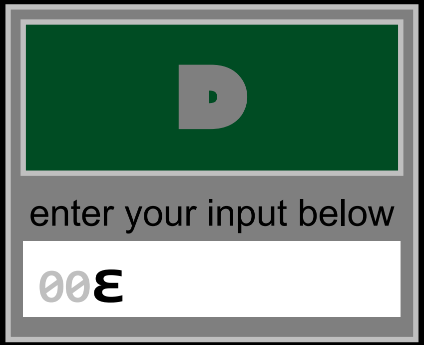
Welche Zustände sind unterscheidbar?

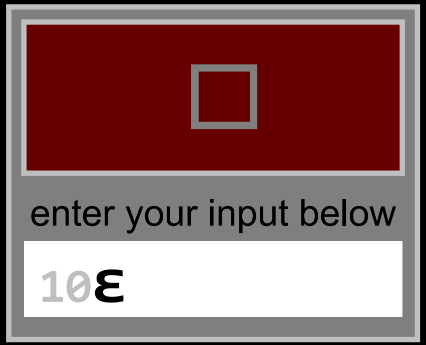








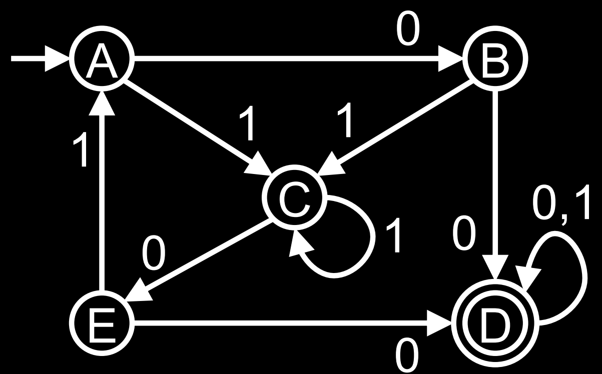




Anschließend hängen wir an alle Instanzen die gleichen Suffixe an.

Im Gegensatz zu allen anderen Instanzen, wird das leere Wort in Automat D akzeptiert. Demzufolge ist der Zustand D von allen anderen unterscheidbar.

Welche Zustände sind unterscheidbar?



A
enter your input below
ε

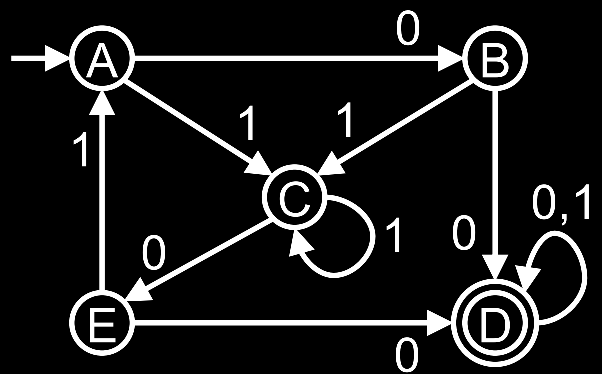
B
enter your input below
0ε

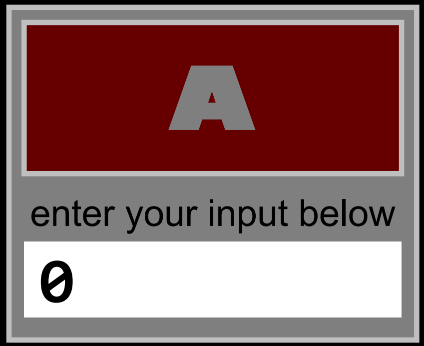
C
enter your input below
1ε

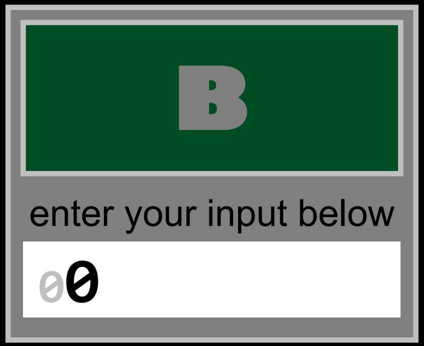
D
enter your input below
00ε

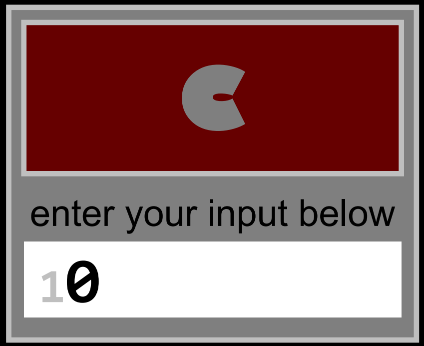
□
enter your input below
10ε

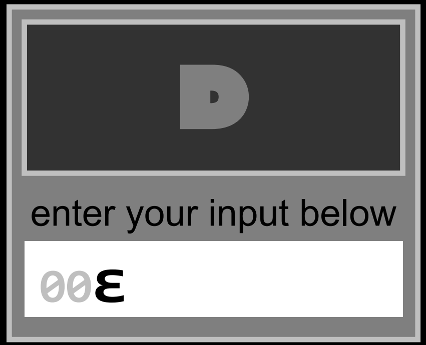
Welche Zustände sind unterscheidbar?



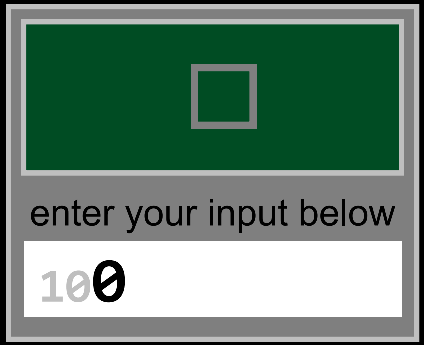




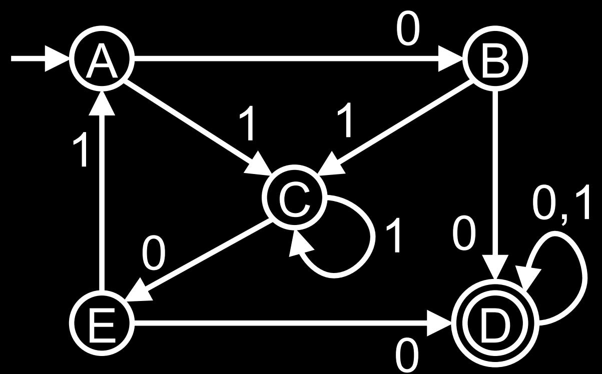




Nun überprüfen wir das Akzeptanzverhalten mit weiteren Worten des Alphabets. D muss nicht weiter beachtet werden, weil er der einzige Repräsentant seiner Äquivalenzklasse ist.



Welche Zustände sind unterscheidbar?



B
enter your input below
00

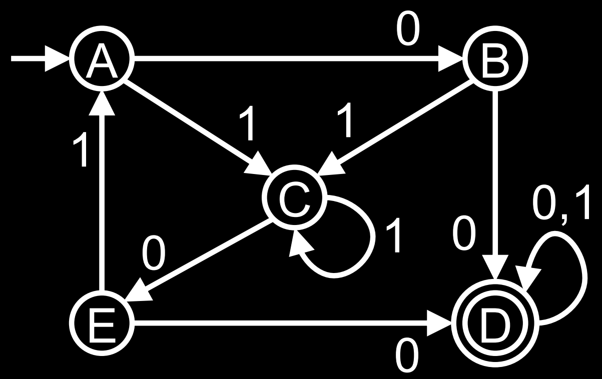
C
enter your input below
10

D
enter your input below
00ε

□
enter your input below
100

A
enter your input below
0

Welche Zustände sind unterscheidbar?



B

enter your input below

01

C

enter your input below

11

D

enter your input below

00ε

□

enter your input below

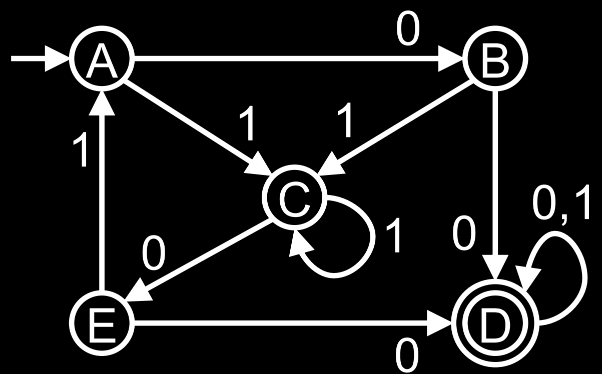
101

A

enter your input below

1

Welche Zustände sind unterscheidbar?



B
enter your input below
000

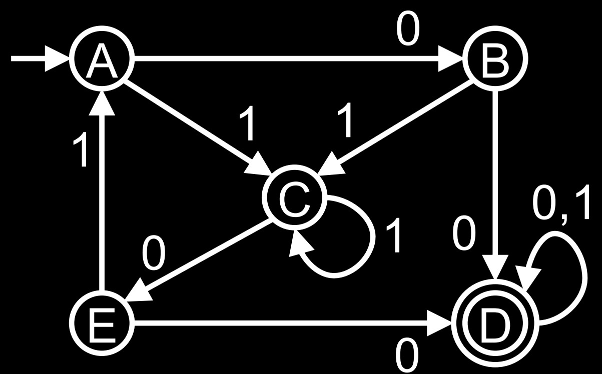
C
enter your input below
100

D
enter your input below
00ε

□
enter your input below
1000

A
enter your input below
00

Welche Zustände sind unterscheidbar?



B

enter your input below

001

C

enter your input below

101

D

enter your input below

00ε

□

enter your input below

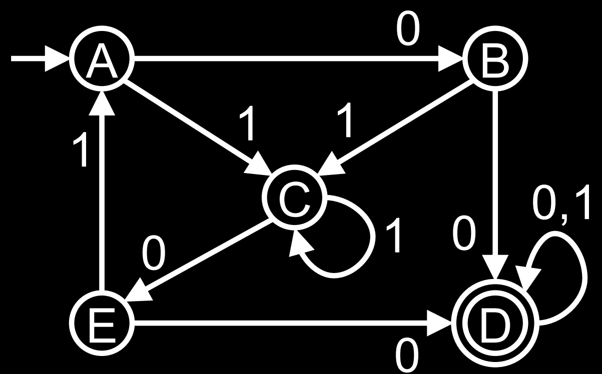
1001

A

enter your input below

01

Welche Zustände sind unterscheidbar?



B

enter your input below

010

C

enter your input below

110

D

enter your input below

00ε

□

enter your input below

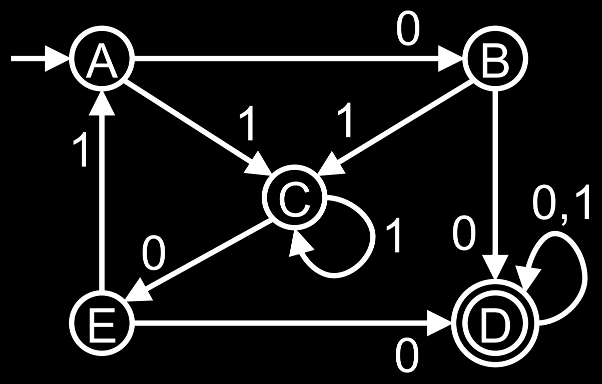
1010

A

enter your input below

10

Welche Zustände sind unterscheidbar?



B
enter your input below
011

C
enter your input below
111

D
enter your input below
00ε

□
enter your input below
1011

A
enter your input below
11

Wie lange müssen wir das nun fortsetzen?

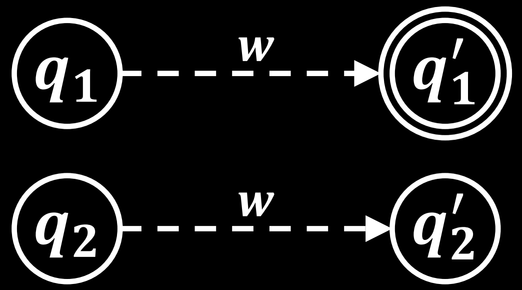
Welche Zustände sind unterscheidbar?

Der bisherige Ansatz ist nicht effizient. Um zu beweisen, dass zwei Zustände nicht unterscheidbar sind, müsste ihr Akzeptanzverhalten für alle Worte des Alphabets identisch sein und somit für alle Worte einzeln geprüft werden. Zudem ist für die Überprüfung jedes einzelnen Wortes eine Ausführung des Automaten notwendig.

Hinweis: Wenn ein Wort aus mehr Symbolen besteht, als der Automat Zustände hat, wird mindestens ein Zustand mehrfach erreicht (Pigeonhole Priniple). Also müssen nur alle Worte mit $|w| < |Q|$ überprüft werden. Es sind aber dennoch zu viele.

Welche Zustände sind unterscheidbar?

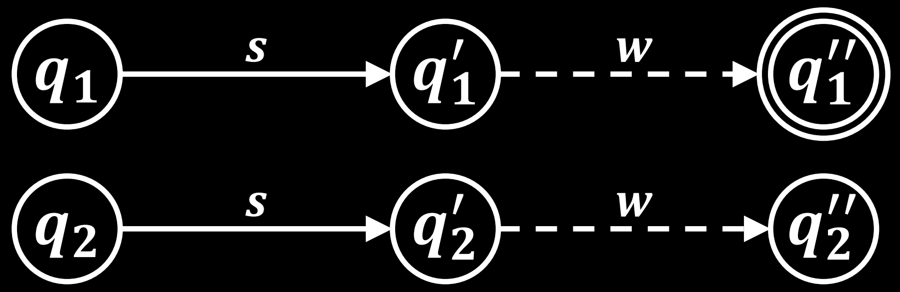
Grundsätzlich müssen wir, um die Unterscheidbarkeit von zwei Zuständen q_1 und q_2 nachzuweisen, **nur ein Wort w** finden, mit dem aus q_1 ein akzeptierender Zustand erreicht wird, aber von q_2 nicht.



Welche Zustände sind unterscheidbar?

Dieser Zusammenhang lässt sich erneut iterativ formulieren: Zwei Zustände q_1 und q_2 sind unterscheidbar, wenn

- q_1 akzeptiert und q_2 nicht – oder umgekehrt
- ein Symbol $s \in \Sigma$ von q_1 und q_2 zu unterscheidbaren Zuständen führt



Was bedeutet Äquivalenz?

Eine Relation $R \subseteq M \times M$ auf einer Menge M ist eine ***Äquivalenzrelation***, falls gilt:

- R ist ***reflexiv***, d.h. für alle $x \in M$ gilt: $(x, x) \in R$.
- R ist ***symmetrisch***, d.h. für alle $x, y \in M$ gilt: Wenn $(x, y) \in R$ ist, dann ist auch $(y, x) \in R$.
- R ist ***transitiv***, d.h. für alle $x, y, z \in M$ gilt: Wenn $(x, y) \in R$ ist und $(y, z) \in R$ ist, dann ist auch $(x, z) \in R$.

Eine Äquivalenzrelation schreibt man häufig auch in Infixnotation und verwendet ein Symbol wie z.B. ≡ (oder ≈). Statt $(x, y) \in R$ schreibt man also etwa $\boldsymbol{x \equiv y}$.

Was bedeutet Äquivalenz?

Durch eine Äquivalenzrelation wird die zu Grunde liegende Menge M in ***Äquivalenzklassen*** eingeteilt. Eine Äquivalenzklasse ist eine Teilmenge von M. Jedes Element x von M liegt in genau einer Äquivalenzklasse, für die man $[x]$ schreibt. Seine Definition lautet: $\boldsymbol{[x] = \{y \mid y \equiv x\}}$.

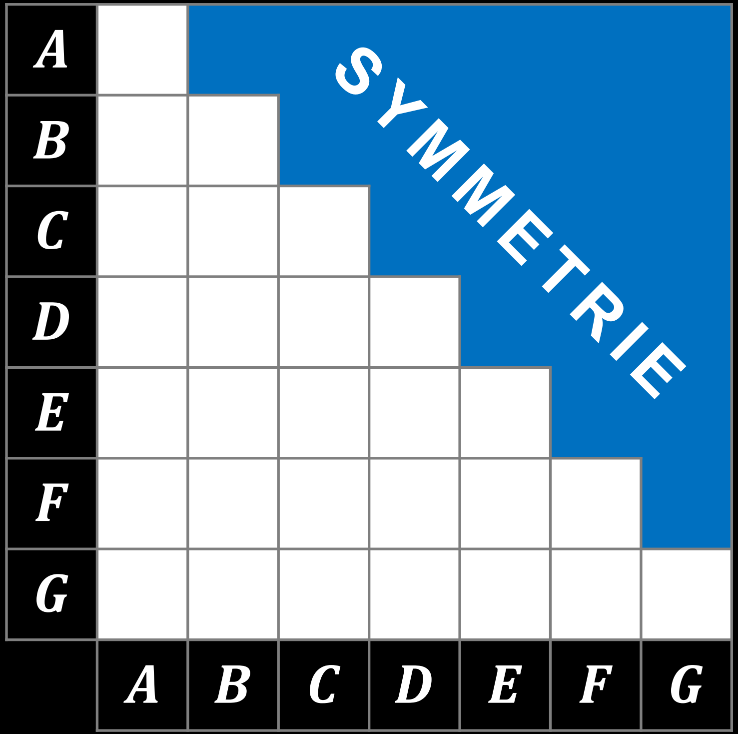
Wenn $x \equiv y$ ist, dann ist $[x] = [y]$ und umgekehrt.

Für die Menge aller Äquivalenzklassen schreibt man auch $M/_{\equiv}$.

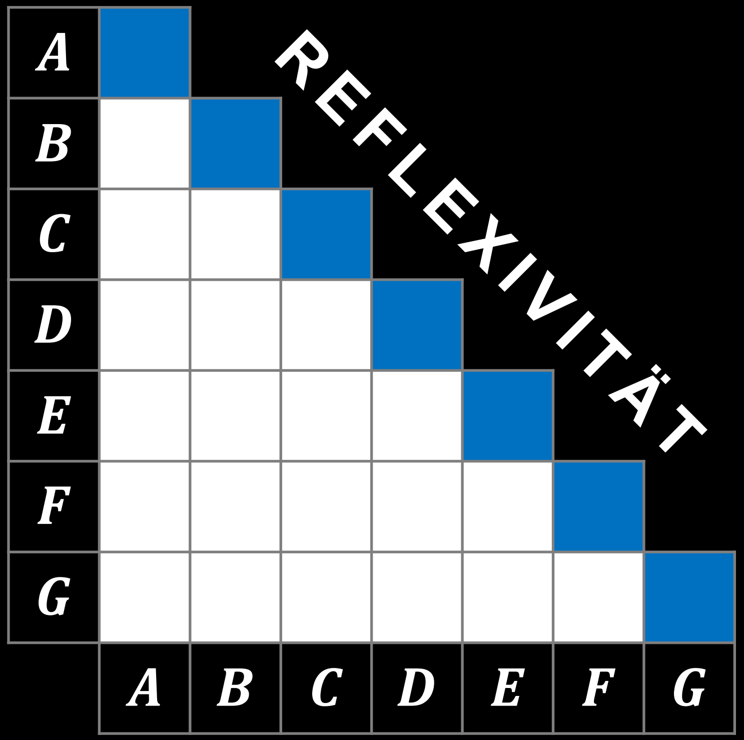
Welche Zustände sind nicht äquivalent?

A							
B							
C							
D							
E							
F							
G							
	A	*B*	*C*	*D*	*E*	*F*	*G*

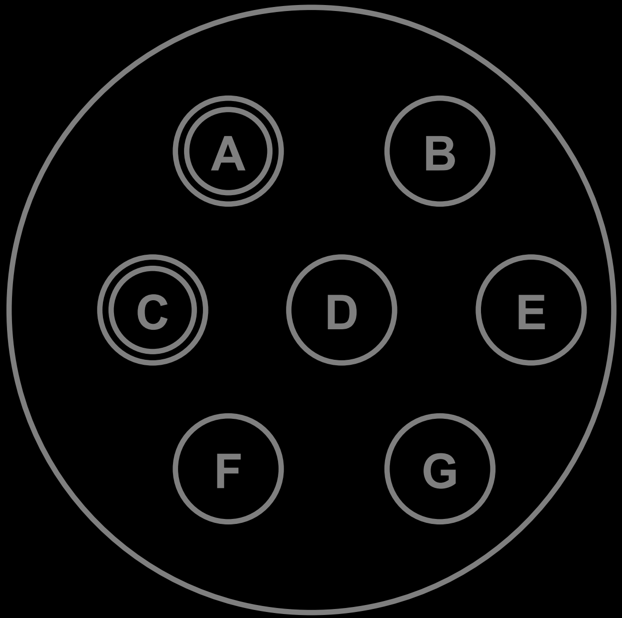
Welche Zustände sind nicht äquivalent?



Welche Zustände sind nicht äquivalent?



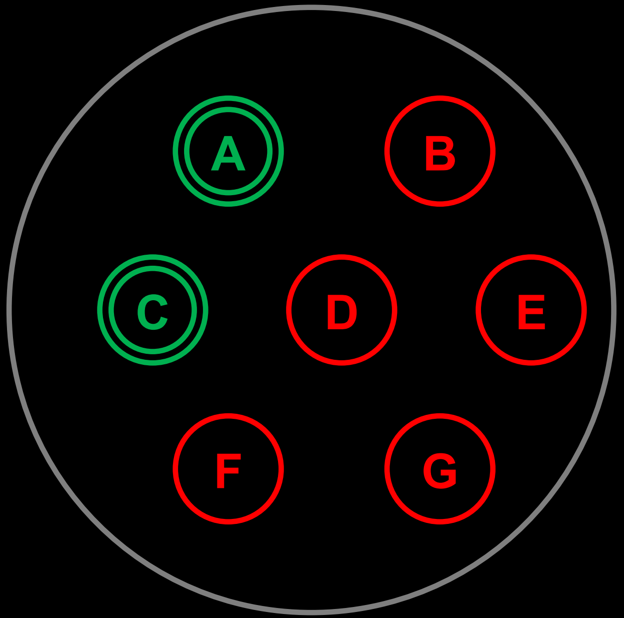
Welche Zustände sind nicht äquivalent?



	A	B	C	D	E	F
B						
C						
D						
E						
F						
G						

δ	A	B	C	D	E	F	G
0	B	C	D	A	F	G	A
1	E	C	D	A	F	G	A

Welche Zustände sind nicht äquivalent?



$$\left.\begin{array}{l}\delta(A,\boldsymbol{\varepsilon}) = \boldsymbol{A} \in \boldsymbol{F} \\ \delta(B,\boldsymbol{\varepsilon}) = \boldsymbol{B} \notin \boldsymbol{F}\end{array}\right\} \boldsymbol{A} \not\equiv_0 \boldsymbol{B}$$

$$\delta(C,\boldsymbol{\varepsilon}) = \boldsymbol{C} \in \boldsymbol{F}$$

$$\left.\begin{array}{l}\delta(D,\boldsymbol{\varepsilon}) = \boldsymbol{D} \notin \boldsymbol{F} \\ \delta(E,\boldsymbol{\varepsilon}) = \boldsymbol{E} \notin \boldsymbol{F}\end{array}\right\} \boldsymbol{D} \equiv_0 \boldsymbol{E}$$

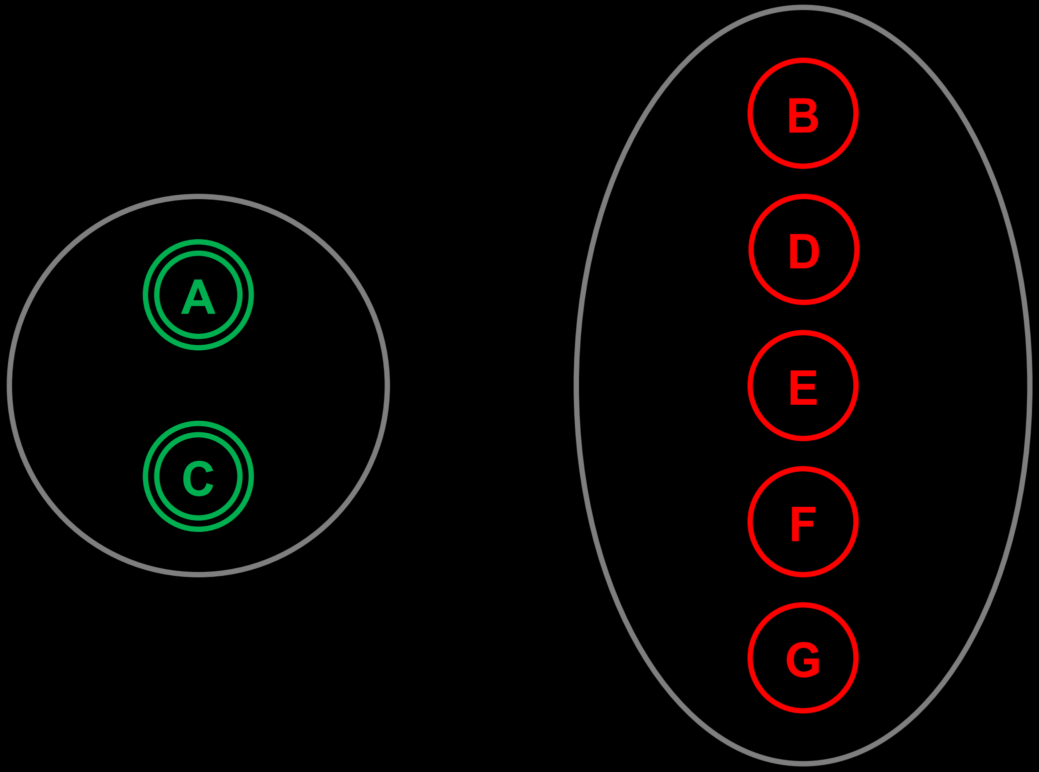
$$\delta(F,\boldsymbol{\varepsilon}) = \boldsymbol{F} \notin \boldsymbol{F}$$

$$\delta(G,\boldsymbol{\varepsilon}) = \boldsymbol{G} \notin \boldsymbol{F}$$

B	$\not\equiv_0$					
C		$\not\equiv_0$				
D	$\not\equiv_0$		$\not\equiv_0$			
E	$\not\equiv_0$		$\not\equiv_0$			
F	$\not\equiv_0$		$\not\equiv_0$			
G	$\not\equiv_0$		$\not\equiv_0$			
	A	B	C	D	E	F

δ	A	B	C	D	E	F	G
0	B	C	D	A	F	G	A
1	E	C	D	A	F	G	A

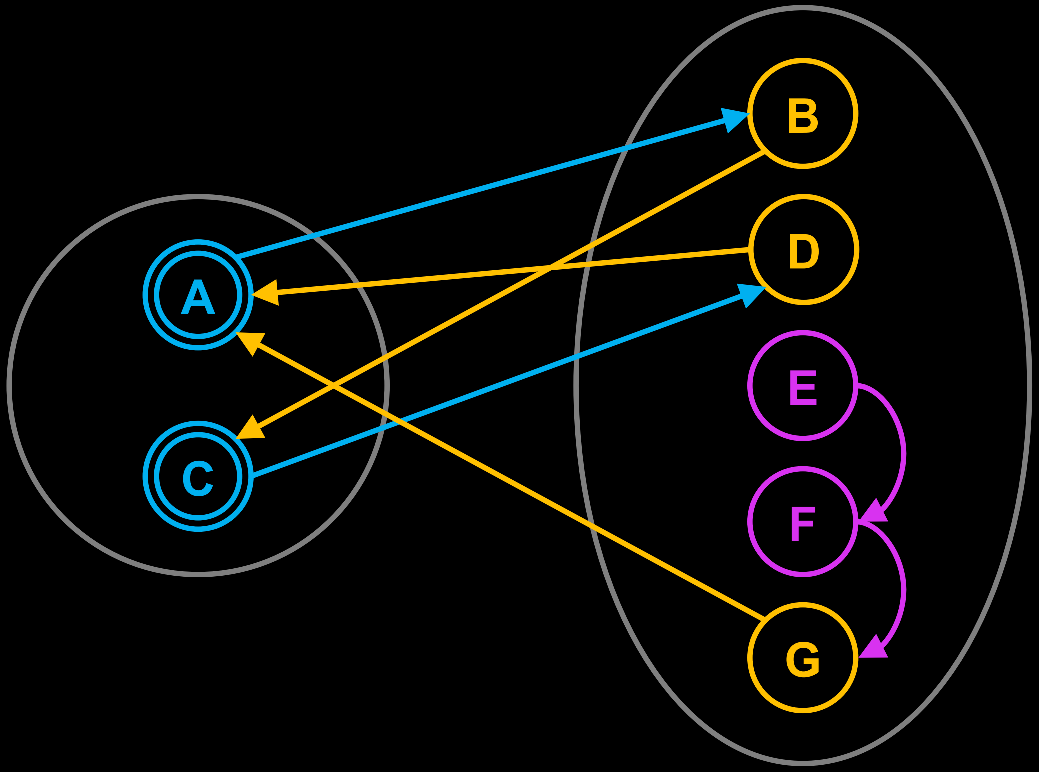
Welche Zustände sind nicht äquivalent?



B	$\neq_0$					
C		$\neq_0$				
D	$\neq_0$		$\neq_0$			
E	$\neq_0$		$\neq_0$			
F	$\neq_0$		$\neq_0$			
G	$\neq_0$		$\neq_0$			
	A	B	C	D	E	F

δ	A	B	C	D	E	F	G
0	B	C	D	A	F	G	A
1	E	C	D	A	F	G	A

Welche Zustände sind nicht äquivalent?

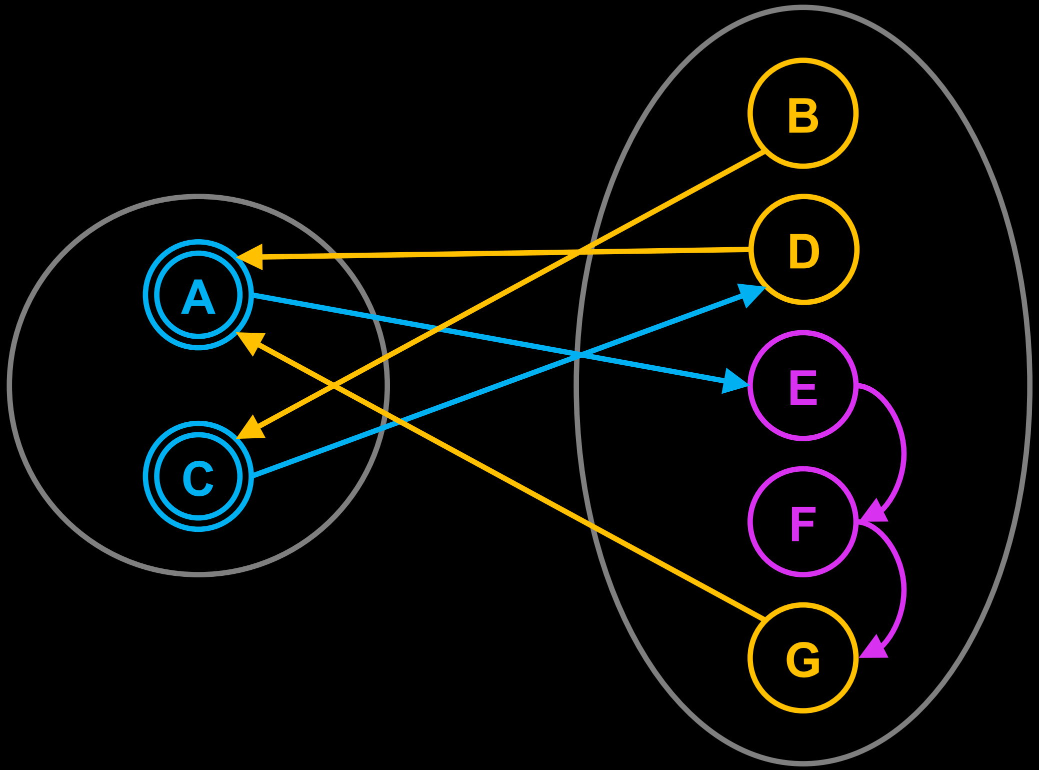


B	$\not\equiv_0$					
C		$\not\equiv_0$				
D	$\not\equiv_0$		$\not\equiv_0$			
E	$\not\equiv_0$	$\not\equiv_1$	$\not\equiv_0$	$\not\equiv_1$		
F	$\not\equiv_0$	$\not\equiv_1$	$\not\equiv_0$	$\not\equiv_1$		
G	$\not\equiv_0$		$\not\equiv_0$		$\not\equiv_1$	$\not\equiv_1$
	A	B	C	D	E	F

δ	A	B	C	D	E	F	G
0	B	C	D	A	F	G	A
1	E	C	D	A	F	G	A

$$\delta(D,0) = A \equiv_0 A = \delta(G,0) \Longrightarrow D \equiv_1 G$$
$$\delta(F,0) = G \not\equiv_0 A = \delta(G,0) \Longrightarrow F \not\equiv_1 G$$
$$\delta(E,0) = F \equiv_0 G = \delta(F,0) \Longrightarrow \cancel{F} \equiv_1 \cancel{G}\ F$$
E

Welche Zustände sind nicht äquivalent?

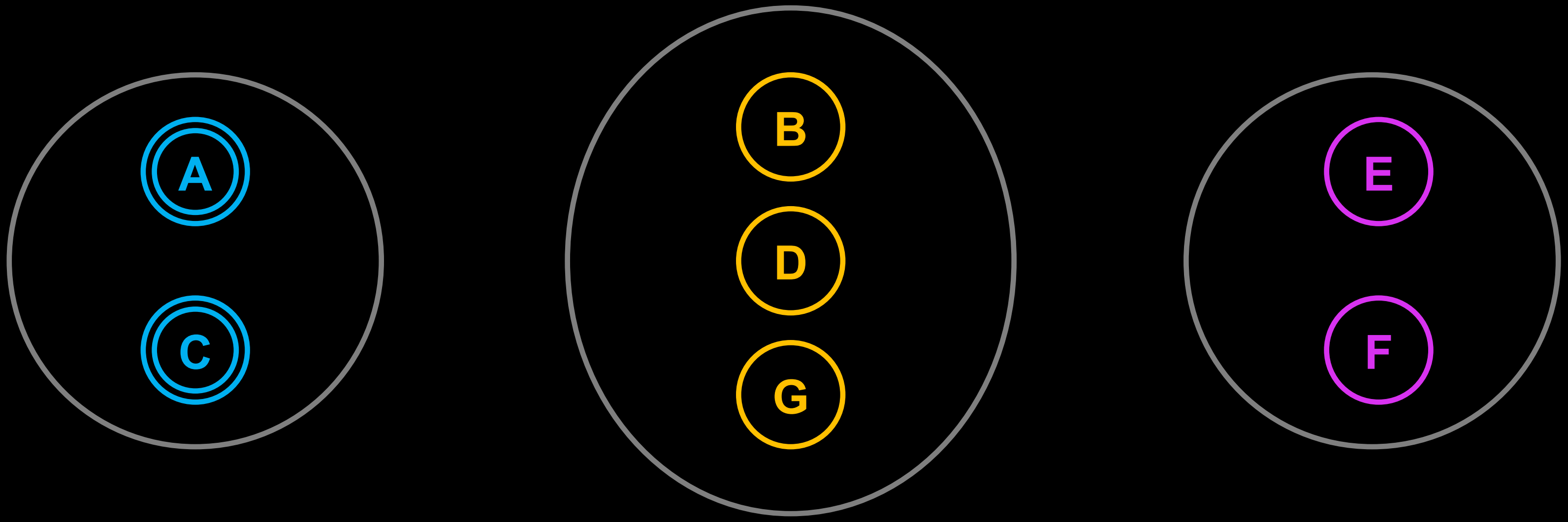


B	$\not\equiv_0$					
C		$\not\equiv_0$				
D	$\not\equiv_0$		$\not\equiv_0$			
E	$\not\equiv_0$	$\not\equiv_1$	$\not\equiv_0$	$\not\equiv_1$		
F	$\not\equiv_0$	$\not\equiv_1$	$\not\equiv_0$	$\not\equiv_1$		
G	$\not\equiv_0$		$\not\equiv_0$		$\not\equiv_1$	$\not\equiv_1$
	A	B	C	D	E	F

$$\delta(D,1) = A \equiv_0 A = \delta(G,1) \Longrightarrow D \equiv_1 G$$
$$\delta(F,1) = G \not\equiv_0 A = \delta(G,1) \Longrightarrow F \not\equiv_1 G$$
$$\delta(E,1) = F \equiv_0 G = \delta(F,1) \Longrightarrow F \equiv_1 G$$

δ	A	B	C	D	E	F	G
0	B	C	D	A	F	G	A
1	E	C	D	A	F	G	A

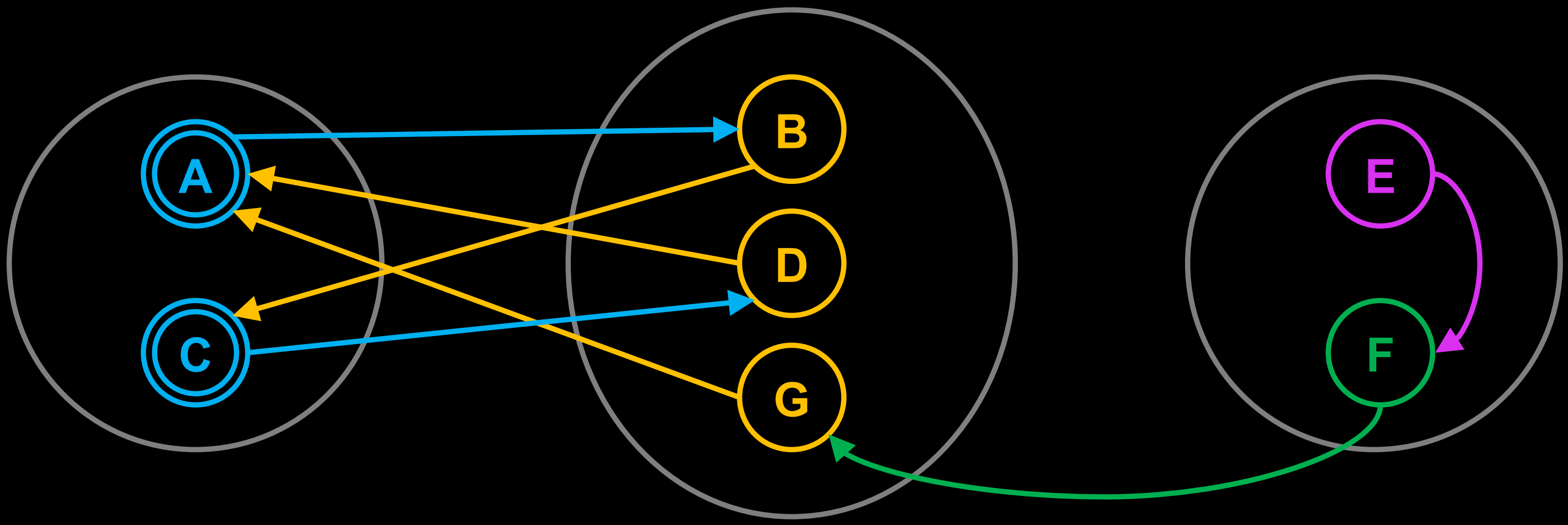
Welche Zustände sind nicht äquivalent?



	A	B	C	D	E	F
B	$\neq_0$					
C		$\neq_0$				
D	$\neq_0$		$\neq_0$			
E	$\neq_0$	$\neq_1$	$\neq_0$	$\neq_1$		
F	$\neq_0$	$\neq_1$	$\neq_0$	$\neq_1$		
G	$\neq_0$		$\neq_0$		$\neq_1$	$\neq_1$

δ	A	B	C	D	E	F	G
0	B	C	D	A	F	G	A
1	E	C	D	A	F	G	A

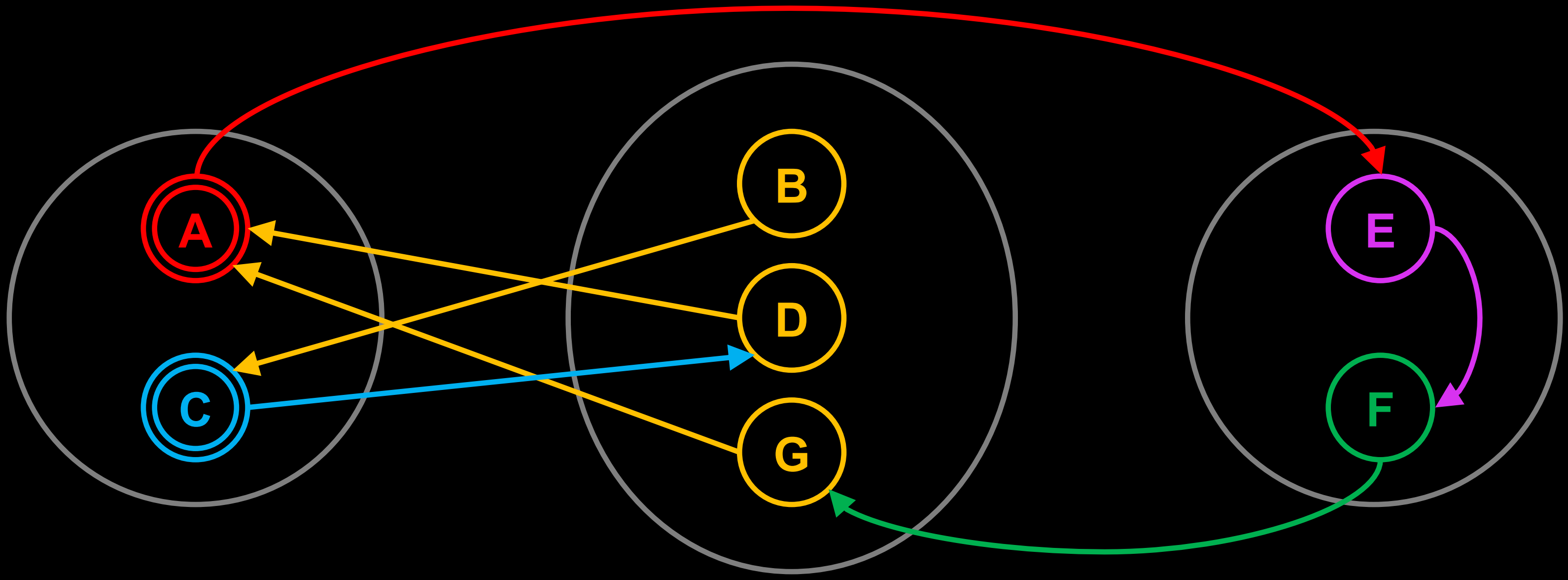
Welche Zustände sind nicht äquivalent?



B	$\neq_0$					
C		$\neq_0$				
D	$\neq_0$		$\neq_0$			
E	$\neq_0$	$\neq_1$	$\neq_0$	$\neq_1$		
F	$\neq_0$	$\neq_1$	$\neq_0$	$\neq_1$	$\neq_2$	
G	$\neq_0$		$\neq_0$		$\neq_1$	$\neq_1$
	A	B	C	D	E	F

δ	A	B	C	D	E	F	G
0	B	C	D	A	F	G	A
1	E	C	D	A	F	G	A

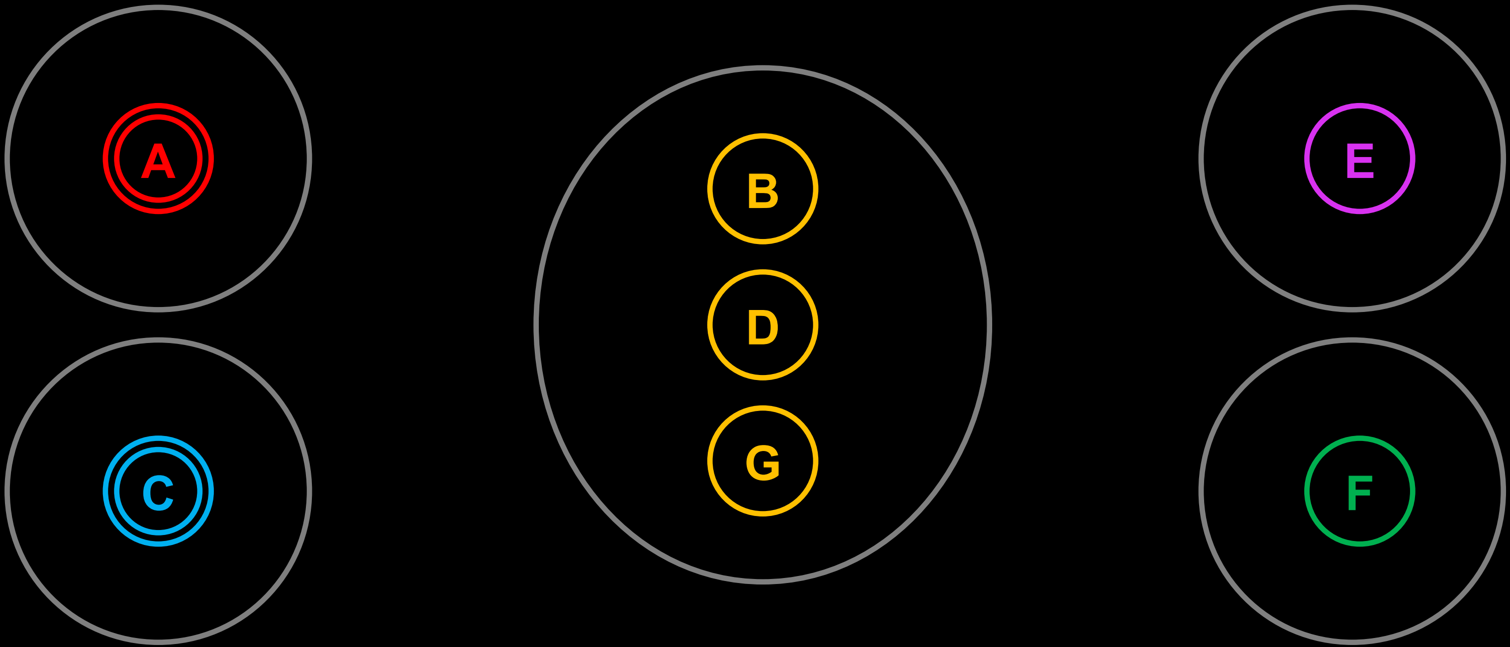
Welche Zustände sind nicht äquivalent?



	A	B	C	D	E	F
B	$\neq_0$					
C	$\neq_2$	$\neq_0$				
D	$\neq_0$		$\neq_0$			
E	$\neq_0$	$\neq_1$	$\neq_0$	$\neq_1$		
F	$\neq_0$	$\neq_1$	$\neq_0$	$\neq_1$	$\neq_2$	
G	$\neq_0$		$\neq_0$		$\neq_1$	$\neq_1$

δ	A	B	C	D	E	F	G
0	B	C	D	A	F	G	A
1	E	C	D	A	F	G	A

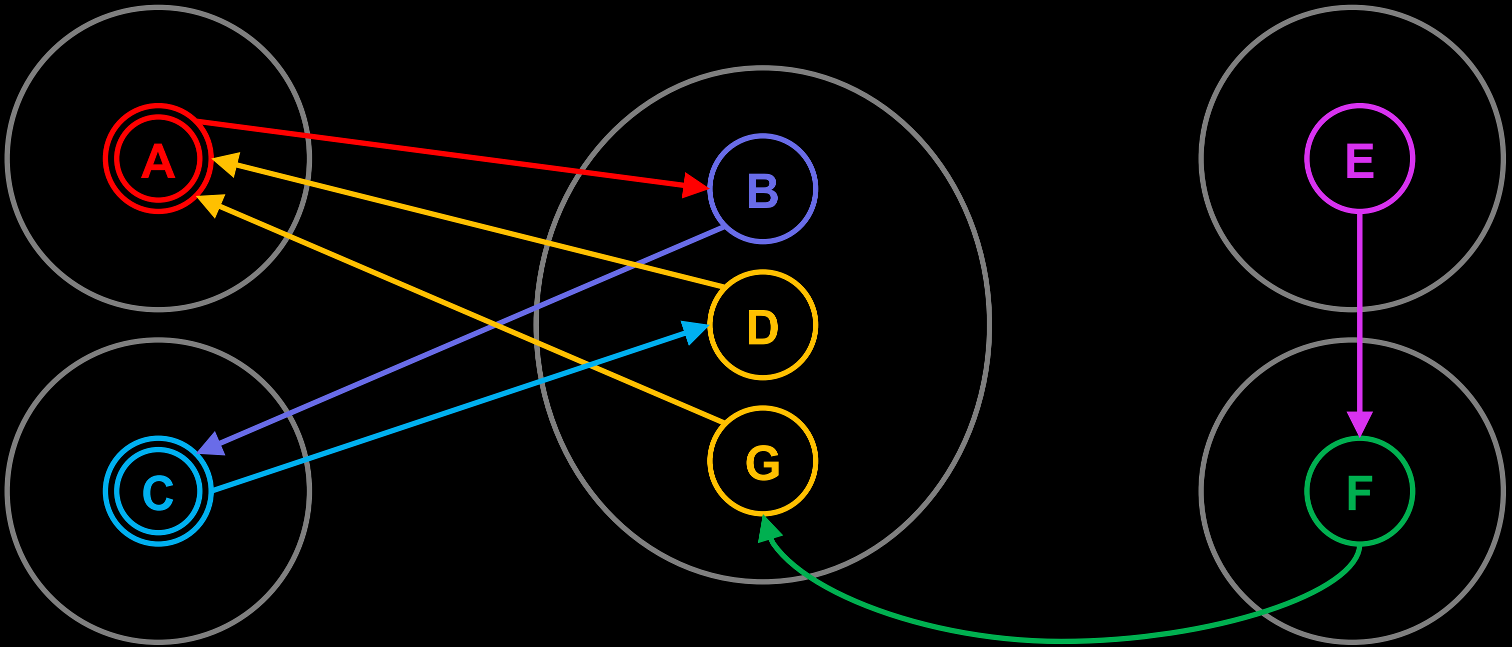
Welche Zustände sind nicht äquivalent?



B	$\neq_0$					
C	$\neq_2$	$\neq_0$				
D	$\neq_0$		$\neq_0$			
E	$\neq_0$	$\neq_1$	$\neq_0$	$\neq_1$		
F	$\neq_0$	$\neq_1$	$\neq_0$	$\neq_1$	$\neq_2$	
G	$\neq_0$		$\neq_0$		$\neq_1$	$\neq_1$
	A	B	C	D	E	F

δ	A	B	C	D	E	F	G
0	B	C	D	A	F	G	A
1	E	C	D	A	F	G	A

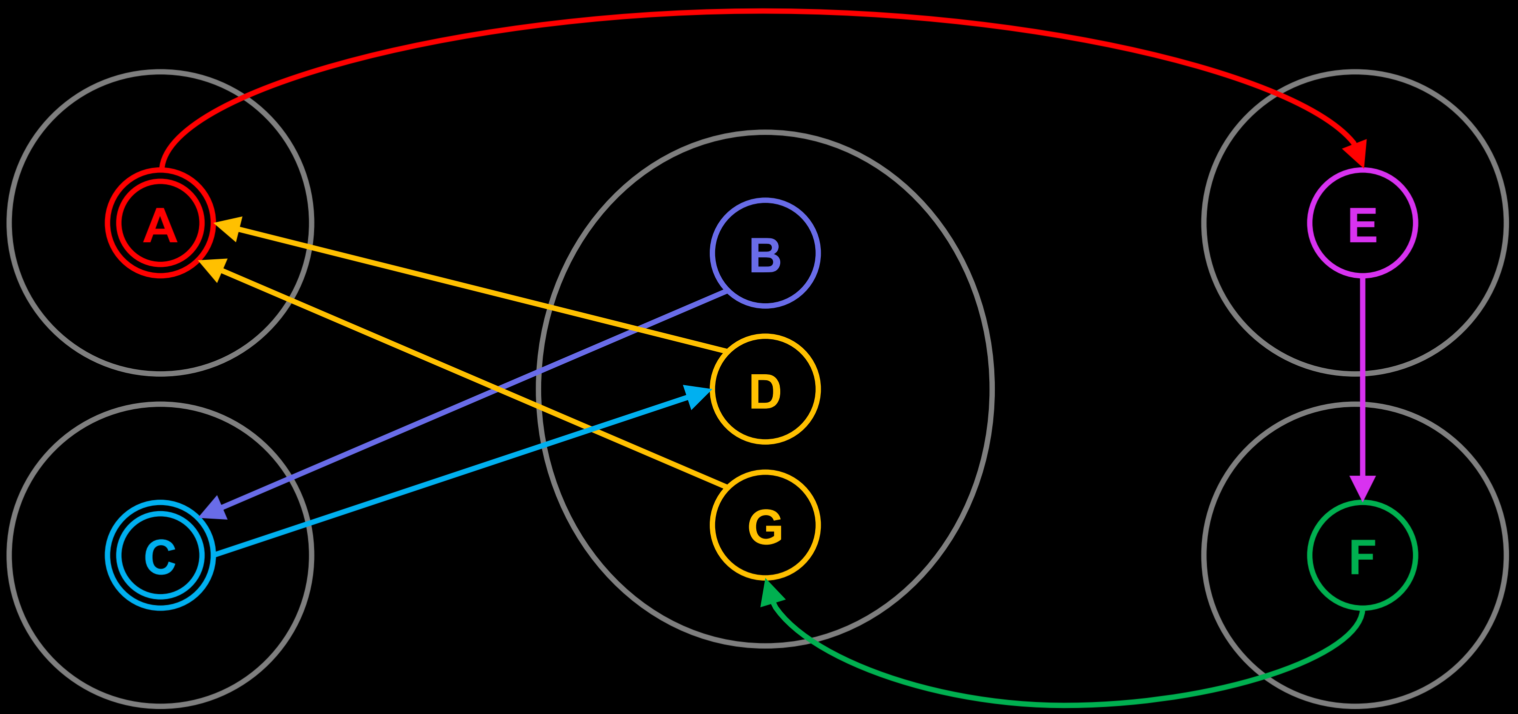
Welche Zustände sind nicht äquivalent?



B	$\neq_0$					
C	$\neq_2$	$\neq_0$				
D	$\neq_0$	$\neq_3$	$\neq_0$			
E	$\neq_0$	$\neq_1$	$\neq_0$	$\neq_1$		
F	$\neq_0$	$\neq_1$	$\neq_0$	$\neq_1$	$\neq_2$	
G	$\neq_0$	$\neq_3$	$\neq_0$		$\neq_1$	$\neq_1$
	A	B	C	D	E	F

δ	A	B	C	D	E	F	G
0	B	C	D	A	F	G	A
1	E	C	D	A	F	G	A

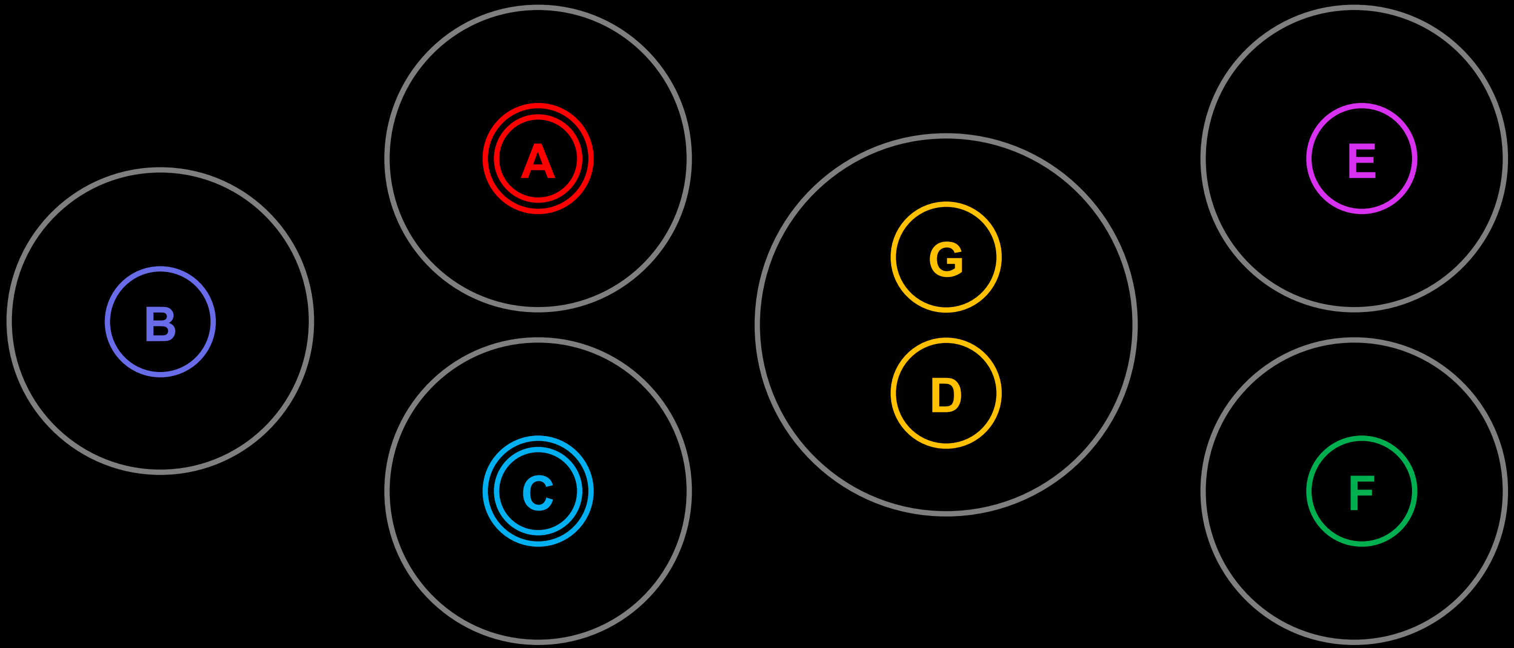
Welche Zustände sind nicht äquivalent?



	A	B	C	D	E	F
B	$\neq_0$					
C	$\neq_2$	$\neq_0$				
D	$\neq_0$	$\neq_3$	$\neq_0$			
E	$\neq_0$	$\neq_1$	$\neq_0$	$\neq_1$		
F	$\neq_0$	$\neq_1$	$\neq_0$	$\neq_1$	$\neq_2$	
G	$\neq_0$	$\neq_3$	$\neq_0$		$\neq_1$	$\neq_1$

δ	A	B	C	D	E	F	G
0	B	C	D	A	F	G	A
1	E	C	D	A	F	G	A

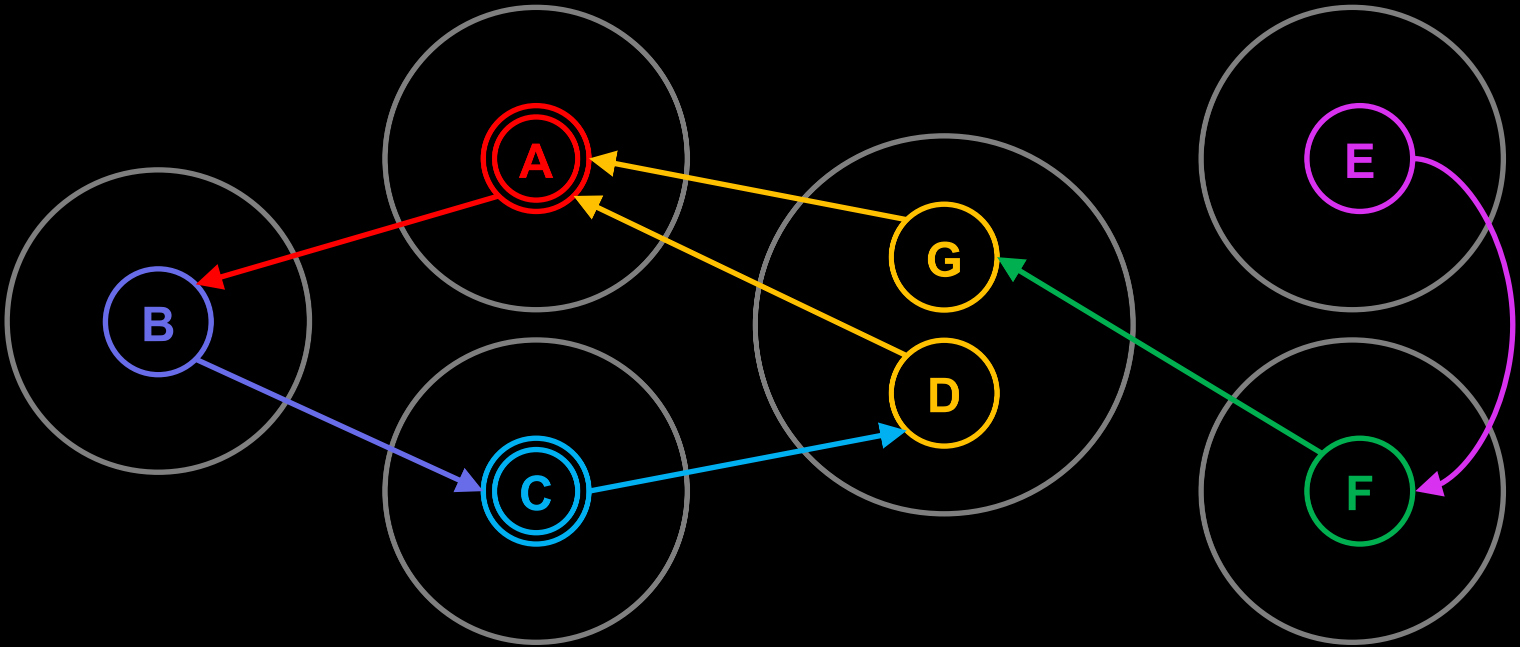
Welche Zustände sind nicht äquivalent?



B	$\neq_0$					
C	$\neq_2$	$\neq_0$				
D	$\neq_0$	$\neq_3$	$\neq_0$			
E	$\neq_0$	$\neq_1$	$\neq_0$	$\neq_1$		
F	$\neq_0$	$\neq_1$	$\neq_0$	$\neq_1$	$\neq_2$	
G	$\neq_0$	$\neq_3$	$\neq_0$		$\neq_1$	$\neq_1$
	A	B	C	D	E	F

δ	A	B	C	D	E	F	G
0	B	C	D	A	F	G	A
1	E	C	D	A	F	G	A

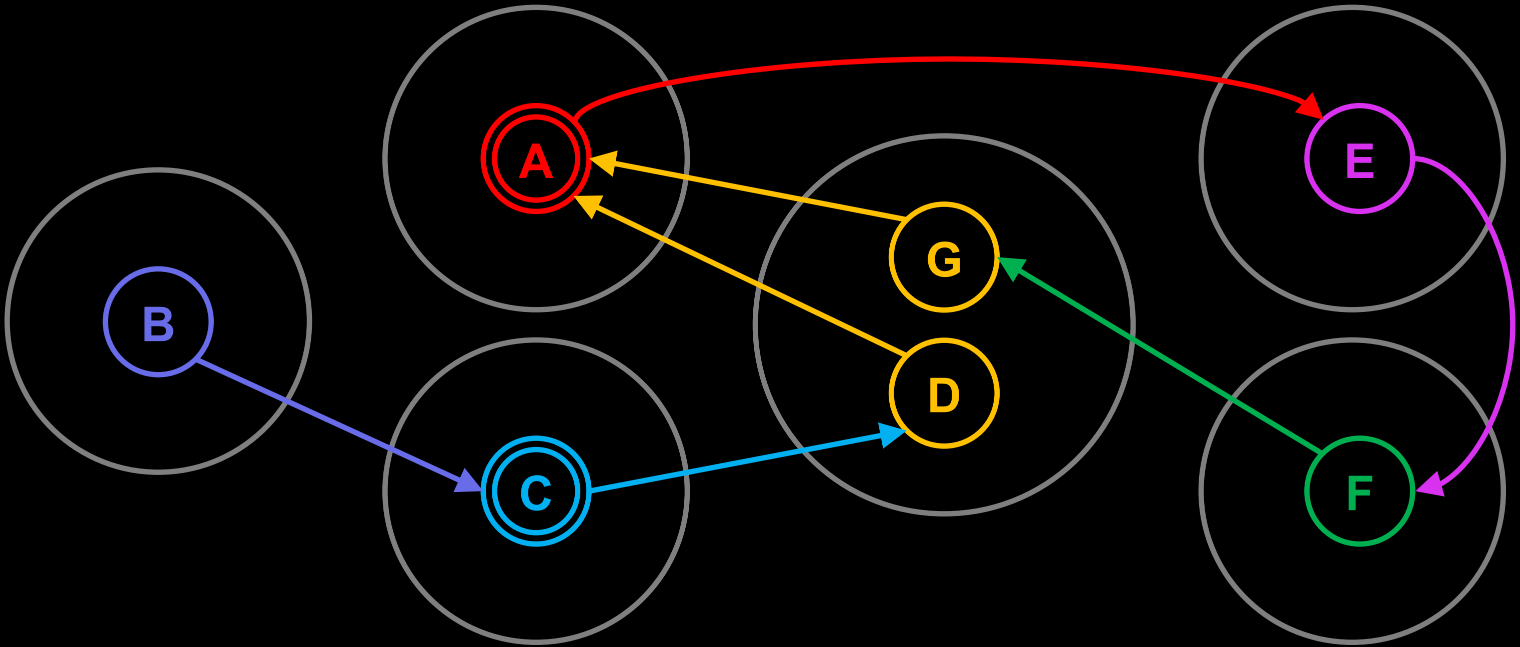
Welche Zustände sind nicht äquivalent?



B	$\neq_0$					
C	$\neq_2$	$\neq_0$				
D	$\neq_0$	$\neq_3$	$\neq_0$			
E	$\neq_0$	$\neq_1$	$\neq_0$	$\neq_1$		
F	$\neq_0$	$\neq_1$	$\neq_0$	$\neq_1$	$\neq_2$	
G	$\neq_0$	$\neq_3$	$\neq_0$		$\neq_1$	$\neq_1$
	A	B	C	D	E	F

δ	A	B	C	D	E	F	G
0	B	C	D	A	F	G	A
1	E	C	D	A	F	G	A

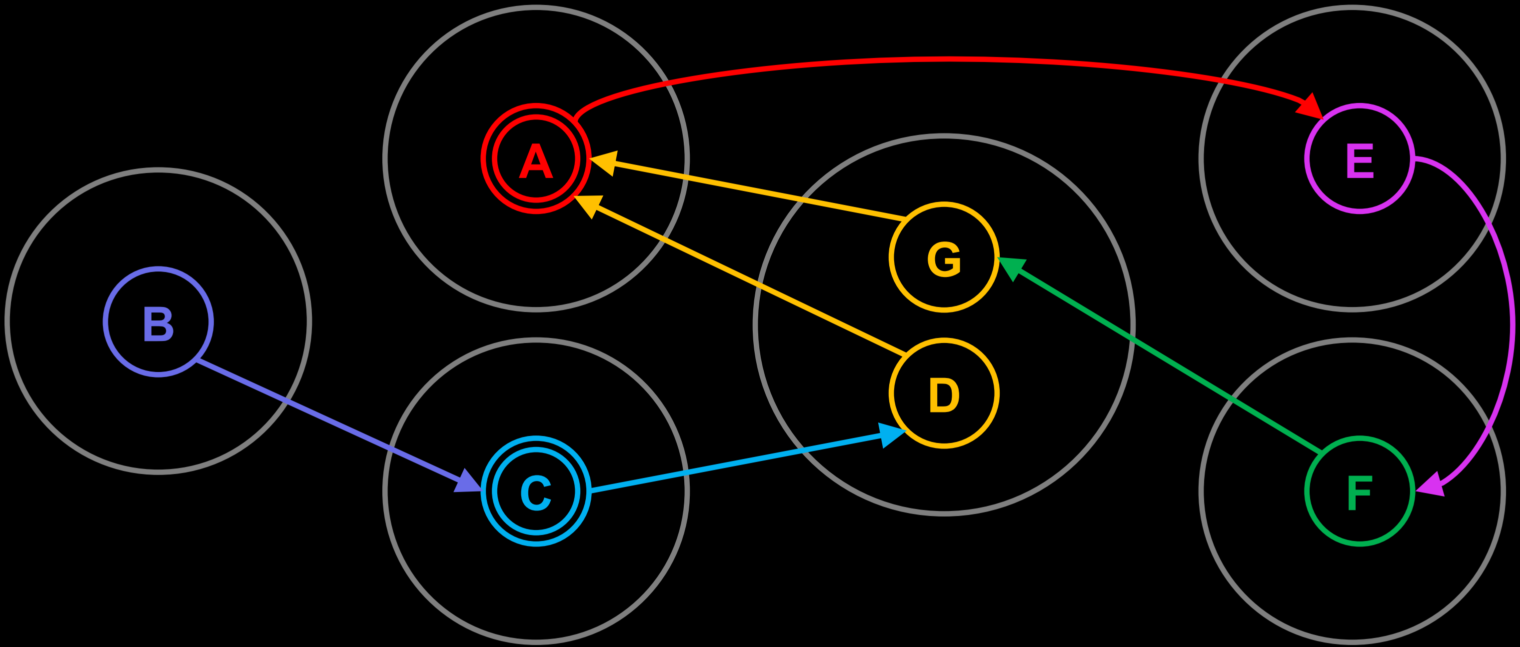
Welche Zustände sind nicht äquivalent?



B	$\neq_0$					
C	$\neq_2$	$\neq_0$				
D	$\neq_0$	$\neq_3$	$\neq_0$			
E	$\neq_0$	$\neq_1$	$\neq_0$	$\neq_1$		
F	$\neq_0$	$\neq_1$	$\neq_0$	$\neq_1$	$\neq_2$	
G	$\neq_0$	$\neq_3$	$\neq_0$		$\neq_1$	$\neq_1$
	A	B	C	D	E	F

δ	A	B	C	D	E	F	G
0	B	C	D	A	F	G	A
1	E	C	D	A	F	G	A

Welche Zustände sind nicht äquivalent?



B	$\neq_0$					
C	$\neq_2$	$\neq_0$				
D	$\neq_0$	$\neq_3$	$\neq_0$			
E	$\neq_0$	$\neq_1$	$\neq_0$	$\neq_1$		
F	$\neq_0$	$\neq_1$	$\neq_0$	$\neq_1$	$\neq_2$	
G	$\neq_0$	$\neq_3$	$\neq_0$		$\neq_1$	$\neq_1$
	A	B	C	D	E	F

Der Algorithmus terminiert jetzt, weil sich bei der letzten Iteration keine Änderung mehr ergeben hat. Eine ausführliche Beschreibung finden Sie im Skript (S. 29ff).

δ	A	B	C	D	E	F	G
0	B	C	D	A	F	G	A
1	E	C	D	A	F	G	A

Wichtige Eigenschaften

Die vorgestellte Zustandsminimierung funktioniert **ausschließlich für deterministische Akzeptoren**! Die Minimierung von nichtdeterministischen Automaten ist deutlich aufwändiger. Das Problem ist PSPACE-vollständig. Ihr werdet bei Herrn Braun noch lernen, was das heißt ;)

Minimale DEA sind **bis auf Isomorphie eindeutig**, d.h. alle minimalen DEA ergeben sich aus dem Äquivalenzklassenautomaten durch Umbenennung der Zustände.