Computability and Complexity
COSC 4200

Deterministic Finite Automata II

Binary Numbers

Each binary string $x \in \{0,1\}^*$ corresponds to a natural number $\mathrm{bin}(x) \in \mathbb{N}$.

x	$\mathrm{bin}(x)$		x	$\mathrm{bin}(x)$
ϵ	0			
0	0		1000	8
1	1		1001	9
10	2		1010	10
11	3		1011	11
100	4		1100	12
101	5		1101	13
110	6		1110	14
111	7		1111	15

Inductive Definition of $\mathrm{bin}(x)$

For any $x \in \{0,1\}^*$, let $\mathrm{bin}(x)$ be the number x encodes in binary. This is defined inductively as follows.

- The base case is $\mathrm{bin}(\epsilon) = 0$.
- For any x, assuming $\mathrm{bin}(x)$ has already been defined,

$$\begin{aligned}
\mathrm{bin}(x0) &= 2\mathrm{bin}(x), \\
\mathrm{bin}(x1) &= 2\mathrm{bin}(x) + 1.
\end{aligned}$$

Inductive Definition of $\mathrm{bin}(x)$

For any $x \in \{0,1\}^*$, let $\mathrm{bin}(x)$ be the number x encodes in binary. This is defined inductively as follows.

- The base case is $\mathrm{bin}(\epsilon) = 0$.
- For any x, assuming $\mathrm{bin}(x)$ has already been defined,

$$\begin{aligned} \mathrm{bin}(x0) &= 2\mathrm{bin}(x), \\ \mathrm{bin}(x1) &= 2\mathrm{bin}(x) + 1. \end{aligned}$$

For example, $\mathrm{bin}(100) = 4$ and

$$\mathrm{bin}(1000) = 2\mathrm{bin}(100) = 2 \cdot 4 = 8,$$

$$\mathrm{bin}(1001) = 2\mathrm{bin}(100) + 1 = 2 \cdot 4 + 1 = 9.$$

Multiples of 3 in Binary

Show that

$$T = \{x \in \{0,1\}^* \mid \text{bin}(x) \text{ is a multiple of } 3\}$$

is regular.

x	$\text{bin}(x)$		x	$\text{bin}(x)$
ϵ	0			
0	0		1000	8
1	1		1001	9
10	2		1010	10
11	3		1011	11
100	4		1100	12
101	5		1101	13
110	6		1110	14
111	7		1111	15

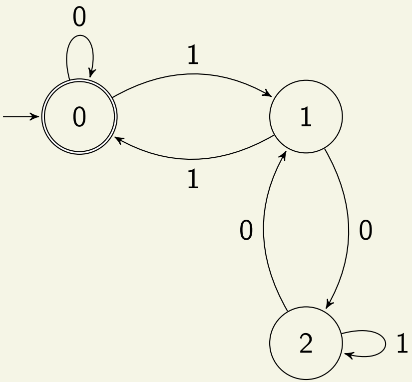
Multiples of 3 in Binary

Show that

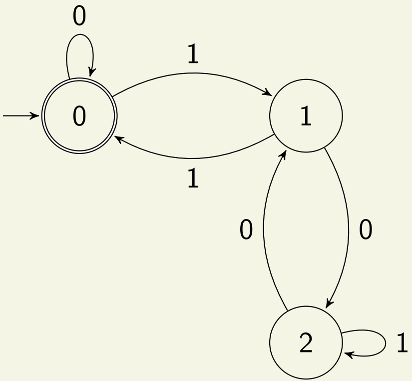
$$T = \{x \in \{0,1\}^* \mid \mathrm{bin}(x) \text{ is a multiple of } 3\}$$

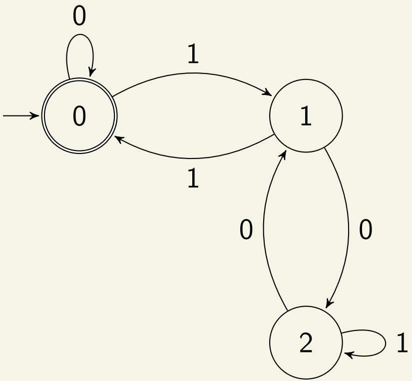
is regular.

x	$\mathrm{bin}(x)$		x	$\mathrm{bin}(x)$
ϵ	0			
0	0		1000	8
1	1		1001	9
10	2		1010	10
11	3		1011	11
100	4		1100	12
101	5		1101	13
110	6		1110	14
111	7		1111	15

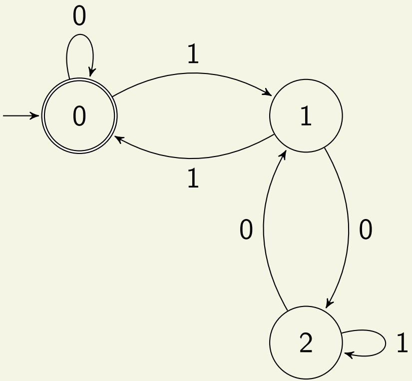


ϵ, 0, 11, 110, 1001, 1100, 1111, ... are accepted


0
1
1
0
1
0
0
2
1



$$Q = \{0,1,2\} \quad \Sigma = \{0,1\}$$
$$\delta : Q \times \Sigma \to Q$$
$$\delta(q,b) = 2q + b \pmod 3$$



$Q = \{0, 1, 2\} \quad \Sigma = \{0, 1\}$
$\delta : Q \times \Sigma \to Q$
$\delta(q, b) = 2q + b \pmod 3$

Claim. For all $x \in \{0,1\}^*$, $\delta^*(0, x) = \mathrm{bin}(x) \pmod 3$.

Induction on Natural Numbers

Goal: Prove $(\forall n \geq 0)\ P(n)$.

Inductive proof:

- **Base case:** Prove $P(0)$.
- **Inductive hypothesis:** Assume that $P(k)$ holds for some $k \geq 0$.
  Inductive step: Use the inductive hypothesis to prove that $P(k+1)$ also holds.

In the proof, we show that $P(0)$ holds and

$$(\forall k \geq 0)\ P(k) \Rightarrow P(k+1).$$

Induction on Binary Strings

Goal: Prove $(\forall x \in \{0,1\}^*)\ P(x)$.

Inductive proof:

- **Base case:** Prove $P(\epsilon)$.
- **Inductive hypothesis:** Assume that $P(x)$ holds for some $x \in \{0,1\}^*$.
  Inductive step: Use the inductive hypothesis to prove that $P(x0)$ and $P(x1)$ also hold.

In the proof, we show that $P(\epsilon)$ holds and

$$(\forall x \in \{0,1\}^*)\ P(x) \Rightarrow P(x0) \wedge P(x1).$$

Claim. For all $x \in \{0,1\}^*$, $\delta^*(0,x) = \mathrm{bin}(x) \pmod 3$.

Proof. (by induction on strings)

Base case: $x = \epsilon$: $\qquad \mathrm{bin}(\epsilon) = 0.$

$$\delta^*(0,\epsilon) = 0 = \mathrm{bin}(\epsilon).$$

Inductive hypothesis: Suppose the claim holds for some $x \in \{0,1\}^*$.

$$\delta^*(0, x) = \text{bin}(x) \pmod 3.$$

Inductive hypothesis: Suppose the claim holds for some $x \in \{0,1\}^*$.

$$\delta^*(0,x) = \operatorname{bin}(x) \pmod 3.$$

Inductive step: We must prove that the claim holds for the extensions $x0$ and $x1$ of x as well:

$$\begin{aligned}
\delta^*(0,x0) &= \operatorname{bin}(x0) \pmod 3 \\
\delta^*(0,x1) &= \operatorname{bin}(x1) \pmod 3
\end{aligned}$$

Inductive hypothesis: Suppose the claim holds for some $x \in \{0,1\}^*$.

$$\delta^*(0, x) = \mathrm{bin}(x) \pmod 3.$$

Inductive step: We must prove that the claim holds for the extensions $x0$ and $x1$ of x as well:

$$\begin{aligned}
\delta^*(0, x0) &= \mathrm{bin}(x0) \pmod 3 \\
\delta^*(0, x1) &= \mathrm{bin}(x1) \pmod 3
\end{aligned}$$

For convenience, we'll work with both statements at the same time:

$$\delta^*(0, xb) = \mathrm{bin}(xb) \pmod 3 \text{ for } b \in \{0,1\}$$

$$\delta^*(0, xb) = \delta(\delta^*(0, x), b) \qquad \text{definition of } \delta^*$$

$$
\begin{array}{rcll}
\delta^*(0, xb) & = & \delta(\delta^*(0, x), b) & \text{definition of } \delta^* \\
& = & 2\delta^*(0, x) + b \pmod 3 & \text{definition of } \delta
\end{array}
$$

$$
\begin{aligned}
\delta^*(0, xb) &= \delta(\delta^*(0, x), b) && \text{definition of } \delta^* \\
&= 2\delta^*(0, x) + b \pmod 3 && \text{definition of } \delta \\
&= 2(\mathrm{bin}(x) \pmod 3) + b \pmod 3 && \text{inductive hypothesis}
\end{aligned}
$$

$$\begin{array}{rcll}
\delta^*(0, xb) & = & \delta(\delta^*(0,x), b) & \text{definition of } \delta^* \\
& = & 2\delta^*(0,x) + b \pmod 3 & \text{definition of } \delta \\
& = & 2(\mathrm{bin}(x) \pmod 3) + b \pmod 3 & \text{inductive hypothesis} \\
& = & 2\mathrm{bin}(x) + b \pmod 3 &
\end{array}$$

$$
\begin{array}{rcll}
\delta^*(0, xb) & = & \delta(\delta^*(0, x), b) & \text{definition of } \delta^* \\
& = & 2\delta^*(0, x) + b \pmod 3 & \text{definition of } \delta \\
& = & 2(\mathrm{bin}(x) \pmod 3) + b \pmod 3 & \text{inductive hypothesis} \\
& = & 2\mathrm{bin}(x) + b \pmod 3 & \\
& = & \mathrm{bin}(xb) \pmod 3 & \text{definition of bin}
\end{array}
$$

$$
\begin{aligned}
\delta^*(0, xb) &= \delta(\delta^*(0, x), b) && \text{definition of } \delta^* \\
&= 2\delta^*(0, x) + b \pmod 3 && \text{definition of } \delta \\
&= 2(\mathrm{bin}(x) \pmod 3) + b \pmod 3 && \text{inductive hypothesis} \\
&= 2\mathrm{bin}(x) + b \pmod 3 && \\
&= \mathrm{bin}(xb) \pmod 3 && \text{definition of bin}
\end{aligned}
$$

Therefore $\delta^*(0, xb) = \mathrm{bin}(xb) \pmod 3$ for both $b \in \{0, 1\}$.

$$
\begin{array}{rcll}
\delta^*(0, xb) & = & \delta(\delta^*(0, x), b) & \text{definition of } \delta^* \\
& = & 2\delta^*(0, x) + b \pmod 3 & \text{definition of } \delta \\
& = & 2(\mathrm{bin}(x) \pmod 3) + b \pmod 3 & \text{inductive hypothesis} \\
& = & 2\mathrm{bin}(x) + b \pmod 3 & \\
& = & \mathrm{bin}(xb) \pmod 3 & \text{definition of bin}
\end{array}
$$

Therefore $\delta^*(0, xb) = \mathrm{bin}(xb) \pmod 3$ for both $b \in \{0, 1\}$.

It follows by mathematical induction that the claim holds for all $x \in \{0, 1\}^*$. $\square$

Regular Operations

We will see that the regular languages are *closed under* the *regular operations* of union, concatenation, and star. This means that if $A, B \in \mathrm{REG}$, then

$$\begin{aligned} A \cup B &\in \mathrm{REG}, \\ A \cdot B &\in \mathrm{REG}, \\ A^* &\in \mathrm{REG}. \end{aligned}$$

$$\begin{aligned} A \cup B &= \{x \mid x \in A \text{ or } x \in B\} \\ A \cdot B &= \{xy \mid x \in A \text{ and } y \in B\} \\ A^* &= \{x_1 x_2 \cdots x_k \mid x_1, x_2, \ldots x_k \in A \text{ and } k \geq 0\} \end{aligned}$$

Theorem

If $A, B \in \mathrm{REG}$, then $A \cup B \in \mathrm{REG}$.

Proof. Let $A, B \in \mathrm{REG}$. Then there exist two DFAs $M_A = (Q_A, \Sigma, \delta_A, q_A, F_A)$ and $M_B = (Q_B, \Sigma, \delta_B, q_B, F_B)$ with $L(M_A) = A$ and $L(M_B) = B$. We will use M_A and M_B to construct a new DFA M with $L(M) = A \cup B$.

Theorem

If $A, B \in \mathrm{REG}$, then $A \cup B \in \mathrm{REG}$.

Proof. Let $A, B \in \mathrm{REG}$. Then there exist two DFAs $M_A = (Q_A, \Sigma, \delta_A, q_A, F_A)$ and $M_B = (Q_B, \Sigma, \delta_B, q_B, F_B)$ with $L(M_A) = A$ and $L(M_B) = B$. We will use M_A and M_B to construct a new DFA M with $L(M) = A \cup B$.

Product Construction. We let $M = (Q, \Sigma, \delta, q_0, F)$ where

- $Q = Q_A \times Q_B = \{(p, q) \mid p \in Q_A, q \in Q_B\}$

Theorem

If $A, B \in \mathrm{REG}$*, then* $A \cup B \in \mathrm{REG}$*.*

Proof. Let $A, B \in \mathrm{REG}$. Then there exist two DFAs $M_A = (Q_A, \Sigma, \delta_A, q_A, F_A)$ and $M_B = (Q_B, \Sigma, \delta_B, q_B, F_B)$ with $L(M_A) = A$ and $L(M_B) = B$. We will use M_A and M_B to construct a new DFA M with $L(M) = A \cup B$.

Product Construction. We let $M = (Q, \Sigma, \delta, q_0, F)$ where

- $Q = Q_A \times Q_B = \{(p, q) \mid p \in Q_A, q \in Q_B\}$
- $\delta : Q \times \Sigma \to Q$ is defined by

$$\delta((p, q), a) = (\delta_A(p, a), \delta_B(q, a))$$

for all $(p, q) \in Q$ and $a \in \Sigma$.

Theorem

If $A, B \in \mathrm{REG}$, then $A \cup B \in \mathrm{REG}$.

Proof. Let $A, B \in \mathrm{REG}$. Then there exist two DFAs $M_A = (Q_A, \Sigma, \delta_A, q_A, F_A)$ and $M_B = (Q_B, \Sigma, \delta_B, q_B, F_B)$ with $L(M_A) = A$ and $L(M_B) = B$. We will use M_A and M_B to construct a new DFA M with $L(M) = A \cup B$.

Product Construction. We let $M = (Q, \Sigma, \delta, q_0, F)$ where

- $Q = Q_A \times Q_B = \{(p, q) \mid p \in Q_A, q \in Q_B\}$
- $\delta : Q \times \Sigma \to Q$ is defined by

$$\delta((p, q), a) = (\delta_A(p, a), \delta_B(q, a))$$

  for all $(p, q) \in Q$ and $a \in \Sigma$.
- $q_0 = (q_A, q_B)$.

Theorem

If $A, B \in \mathrm{REG}$, then $A \cup B \in \mathrm{REG}$.

Proof. Let $A, B \in \mathrm{REG}$. Then there exist two DFAs $M_A = (Q_A, \Sigma, \delta_A, q_A, F_A)$ and $M_B = (Q_B, \Sigma, \delta_B, q_B, F_B)$ with $L(M_A) = A$ and $L(M_B) = B$. We will use M_A and M_B to construct a new DFA M with $L(M) = A \cup B$.

Product Construction. We let $M = (Q, \Sigma, \delta, q_0, F)$ where

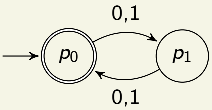
- $Q = Q_A \times Q_B = \{(p, q) \mid p \in Q_A, q \in Q_B\}$
- $\delta : Q \times \Sigma \to Q$ is defined by

  $$\delta((p, q), a) = (\delta_A(p, a), \delta_B(q, a))$$

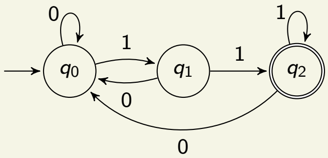
  for all $(p, q) \in Q$ and $a \in \Sigma$.
- $q_0 = (q_A, q_B)$.
- $F = \{(p, q) \in Q \mid p \in F_A \text{ or } q \in F_B\}$

Example - Product Construction

$A = \{x \in \{0,1\}^* \mid |x| \text{ is even}\}$

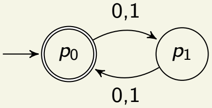


$B = \{x11 \mid x \in \{0,1\}^*\}$

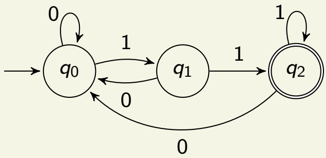


Example - Product Construction

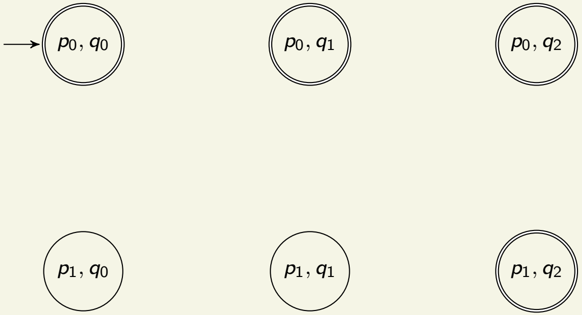
$A = \{x \in \{0,1\}^* \mid |x| \text{ is even}\}$



$B = \{x11 \mid x \in \{0,1\}^*\}$

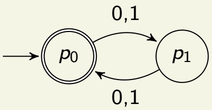


$A \cup B$:

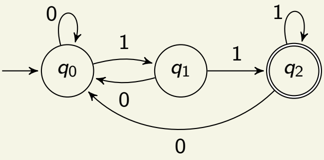


Example - Product Construction

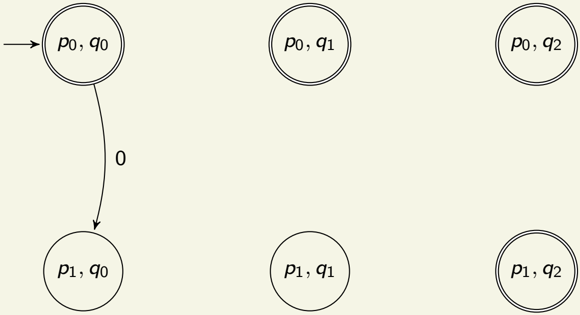
$A = \{x \in \{0,1\}^* \mid |x| \text{ is even}\}$



$B = \{x11 \mid x \in \{0,1\}^*\}$

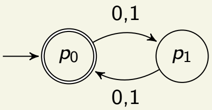


$A \cup B$:

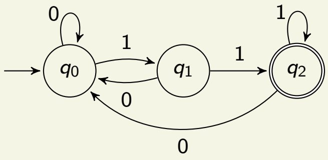


Example - Product Construction

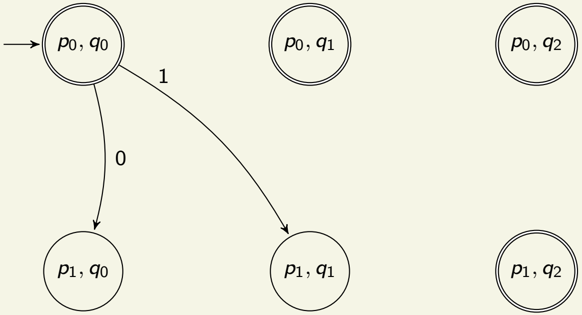
$A = \{x \in \{0,1\}^* \mid |x| \text{ is even}\}$



$B = \{x11 \mid x \in \{0,1\}^*\}$

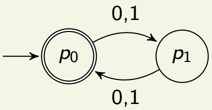


$A \cup B$:

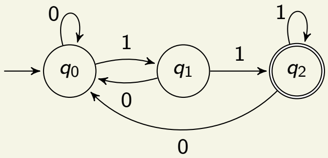


Example - Product Construction

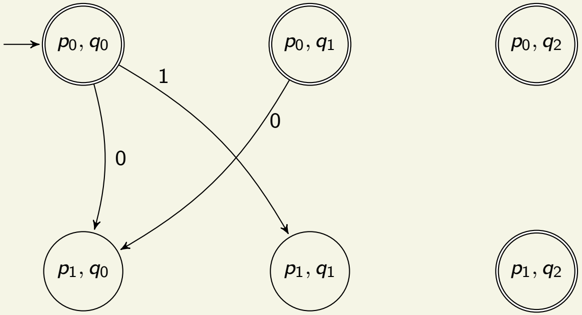
$A = \{x \in \{0,1\}^* \mid |x| \text{ is even}\}$



$B = \{x11 \mid x \in \{0,1\}^*\}$

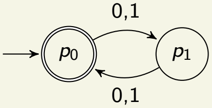


$A \cup B$:

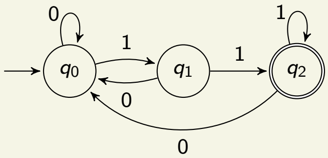


Example - Product Construction

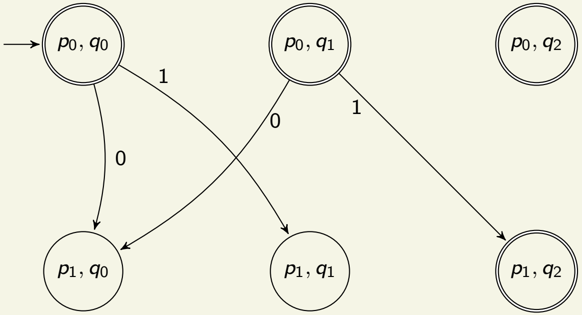
$A = \{x \in \{0,1\}^* \mid |x| \text{ is even}\}$



$B = \{x11 \mid x \in \{0,1\}^*\}$

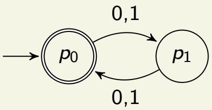


$A \cup B$:

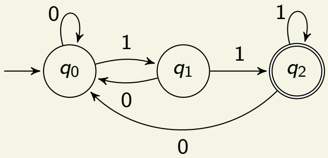


Example - Product Construction

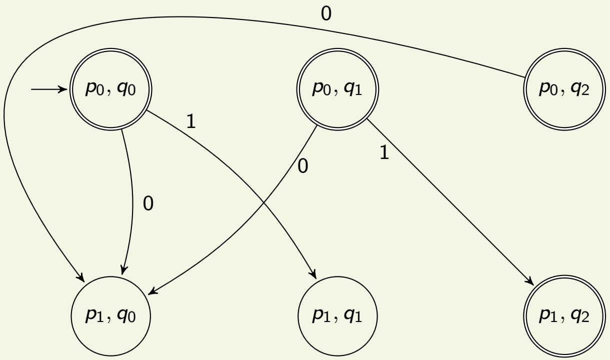
$A = \{x \in \{0,1\}^* \mid |x| \text{ is even}\}$



$B = \{x11 \mid x \in \{0,1\}^*\}$

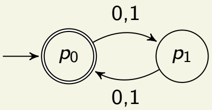


$A \cup B$:

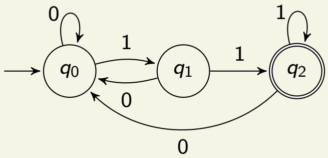


Example - Product Construction

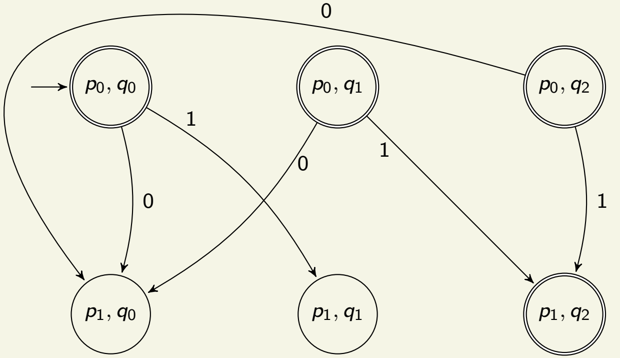
$A = \{x \in \{0,1\}^* \mid |x| \text{ is even}\}$



$B = \{x11 \mid x \in \{0,1\}^*\}$

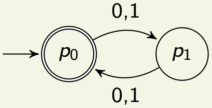


$A \cup B$:

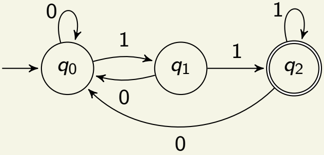


Example - Product Construction

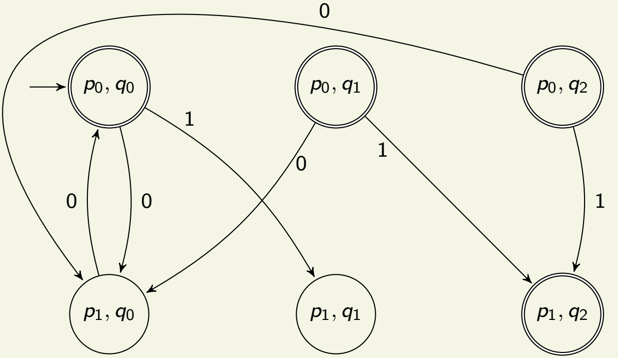
$A = \{x \in \{0,1\}^* \mid |x| \text{ is even}\}$



$B = \{x11 \mid x \in \{0,1\}^*\}$

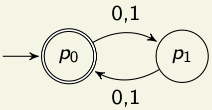


$A \cup B$:

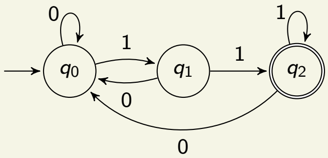


Example - Product Construction

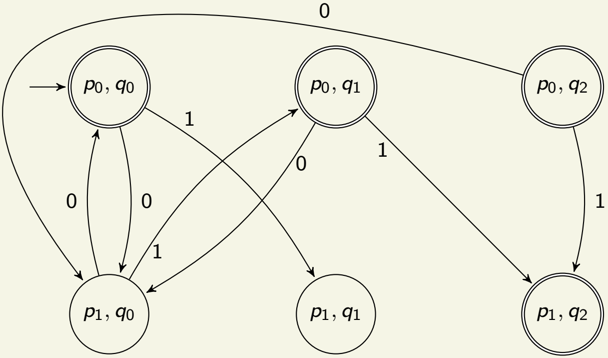
$A = \{x \in \{0,1\}^* \mid |x| \text{ is even}\}$



$B = \{x11 \mid x \in \{0,1\}^*\}$

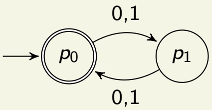


$A \cup B$:

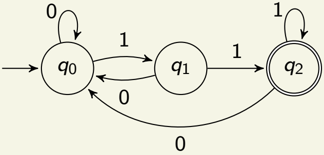


Example - Product Construction

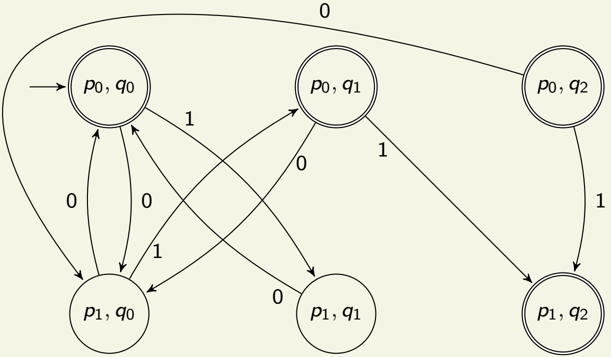
$A = \{x \in \{0,1\}^* \mid |x| \text{ is even}\}$



$B = \{x11 \mid x \in \{0,1\}^*\}$

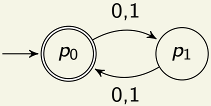


$A \cup B$:

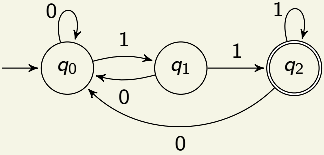


Example - Product Construction

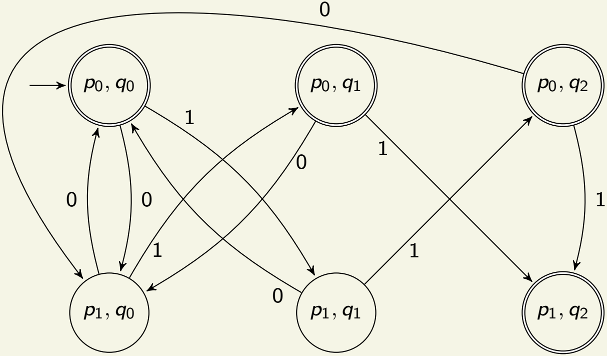
$A = \{x \in \{0,1\}^* \mid |x| \text{ is even}\}$



$B = \{x11 \mid x \in \{0,1\}^*\}$

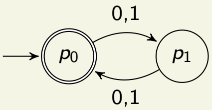


$A \cup B$:

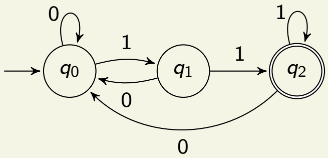


Example - Product Construction

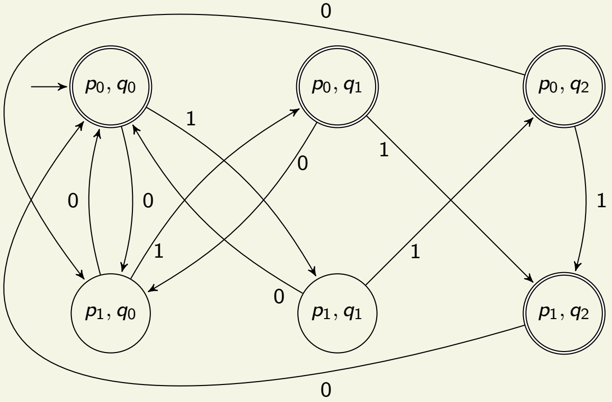
$A = \{x \in \{0,1\}^* \mid |x| \text{ is even}\}$



$B = \{x11 \mid x \in \{0,1\}^*\}$

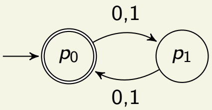


$A \cup B$:

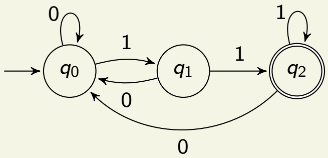


Example - Product Construction

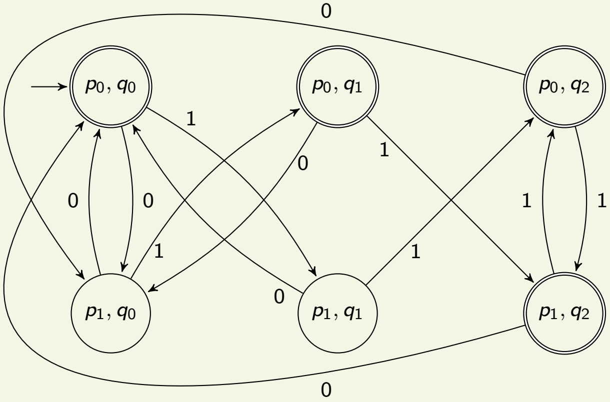
$A = \{x \in \{0,1\}^* \mid |x| \text{ is even}\}$



$B = \{x11 \mid x \in \{0,1\}^*\}$



$A \cup B$:



Claim. For all $x \in \Sigma^*$, $\delta^*(q_0, x) = (\delta_A^*(q_A, x), \delta_B^*(q_B, x))$.

Proof. The proof is by induction on strings.

For the base case, we have

$$\delta^*(q_0, \epsilon) = q_0 = (q_A, q_B) = (\delta^*(q_A, \epsilon), \delta^*(q_B, \epsilon)).$$

Claim. For all $x \in \Sigma^*$, $\delta^*(q_0, x) = (\delta_A^*(q_A, x), \delta_B^*(q_B, x))$.

Proof. The proof is by induction on strings.

For the base case, we have

$$\delta^*(q_0, \epsilon) = q_0 = (q_A, q_B) = (\delta^*(q_A, \epsilon), \delta^*(q_B, \epsilon)).$$

Now suppose the claim holds for some string x. For the inductive step, we must show the claim also holds for $x0$ and $x1$. For all $b \in \{0, 1\}$,

Claim. For all $x \in \Sigma^*$, $\delta^*(q_0, x) = (\delta_A^*(q_A, x), \delta_B^*(q_B, x))$.

Proof. The proof is by induction on strings.

For the base case, we have

$$\delta^*(q_0, \epsilon) = q_0 = (q_A, q_B) = (\delta^*(q_A, \epsilon), \delta^*(q_B, \epsilon)).$$

Now suppose the claim holds for some string x. For the inductive step, we must show the claim also holds for $x0$ and $x1$. For all $b \in \{0, 1\}$,

$$\delta^*(q_0, xb) \quad = \quad \delta(\delta^*(q_0, x), b)$$

Claim. For all $x \in \Sigma^*$, $\delta^*(q_0, x) = (\delta_A^*(q_A, x), \delta_B^*(q_B, x))$.

Proof. The proof is by induction on strings.

For the base case, we have

$$\delta^*(q_0, \epsilon) = q_0 = (q_A, q_B) = (\delta^*(q_A, \epsilon), \delta^*(q_B, \epsilon)).$$

Now suppose the claim holds for some string x. For the inductive step, we must show the claim also holds for $x0$ and $x1$. For all $b \in \{0, 1\}$,

$$\begin{aligned} \delta^*(q_0, xb) &= \delta(\delta^*(q_0, x), b) \\ &= \delta((\delta_A^*(q_A, x), \delta_B^*(q_B, x)), b) \end{aligned}$$

Claim. For all $x \in \Sigma^*$, $\delta^*(q_0, x) = (\delta_A^*(q_A, x), \delta_B^*(q_B, x))$.

Proof. The proof is by induction on strings.

For the base case, we have

$$\delta^*(q_0, \epsilon) = q_0 = (q_A, q_B) = (\delta^*(q_A, \epsilon), \delta^*(q_B, \epsilon)).$$

Now suppose the claim holds for some string x. For the inductive step, we must show the claim also holds for $x0$ and $x1$. For all $b \in \{0, 1\}$,

$$\begin{aligned}
\delta^*(q_0, xb) &= \delta(\delta^*(q_0, x), b) \\
&= \delta((\delta_A^*(q_A, x), \delta_B^*(q_B, x)), b) \\
&= (\delta_A(\delta_A^*(q_A, x), b), \delta_B(\delta_B^*(q_B, x), b))
\end{aligned}$$

Claim. For all $x \in \Sigma^*$, $\delta^*(q_0, x) = (\delta_A^*(q_A, x), \delta_B^*(q_B, x))$.

Proof. The proof is by induction on strings.

For the base case, we have

$$\delta^*(q_0, \epsilon) = q_0 = (q_A, q_B) = (\delta^*(q_A, \epsilon), \delta^*(q_B, \epsilon)).$$

Now suppose the claim holds for some string x. For the inductive step, we must show the claim also holds for $x0$ and $x1$. For all $b \in \{0, 1\}$,

$$\begin{aligned}
\delta^*(q_0, xb) &= \delta(\delta^*(q_0, x), b) \\
&= \delta((\delta_A^*(q_A, x), \delta_B^*(q_B, x)), b) \\
&= (\delta_A(\delta_A^*(q_A, x), b), \delta_B(\delta_B^*(q_B, x), b)) \\
&= (\delta_A^*(q_A, xb), \delta_B^*(q_B, xb)). \quad \square
\end{aligned}$$

For any $x \in \Sigma^*$, we have

$$x \in L(M) \quad \Leftrightarrow \quad \delta^*(q_0, x) \in F$$

For any $x \in \Sigma^*$, we have

$$\begin{aligned} x \in L(M) &\Leftrightarrow \delta^*(q_0, x) \in F \\ &\Leftrightarrow (\delta_A^*(q_A, x), \delta_B^*(q_B, x)) \in F \end{aligned}$$

For any $x \in \Sigma^*$, we have

$$\begin{aligned} x \in L(M) &\Leftrightarrow \delta^*(q_0, x) \in F \\ &\Leftrightarrow (\delta_A^*(q_A, x), \delta_B^*(q_B, x)) \in F \\ &\Leftrightarrow \delta_A^*(q_A, x) \in F_A \text{ or } \delta_B^*(q_B, x)) \in F_B \end{aligned}$$

For any $x \in \Sigma^*$, we have

$$
\begin{aligned}
x \in L(M) &\Leftrightarrow \delta^*(q_0, x) \in F \\
&\Leftrightarrow (\delta_A^*(q_A, x), \delta_B^*(q_B, x)) \in F \\
&\Leftrightarrow \delta_A^*(q_A, x) \in F_A \text{ or } \delta_B^*(q_B, x)) \in F_B \\
&\Leftrightarrow x \in L(M_A) \text{ or } x \in L(M_B)
\end{aligned}
$$

For any $x \in \Sigma^*$, we have

$$\begin{aligned}
x \in L(M) &\Leftrightarrow \delta^*(q_0, x) \in F \\
&\Leftrightarrow (\delta_A^*(q_A, x), \delta_B^*(q_B, x)) \in F \\
&\Leftrightarrow \delta_A^*(q_A, x) \in F_A \text{ or } \delta_B^*(q_B, x)) \in F_B \\
&\Leftrightarrow x \in L(M_A) \text{ or } x \in L(M_B) \\
&\Leftrightarrow x \in L(M_A) \cup L(M_B) = A \cup B.
\end{aligned}$$

Therefore $L(M) = A \cup B$, so $A \cup B$ is regular. $\square$

Closure Under Complementation

The *complement* of a language A is

$$A^c = \{x \in \Sigma^* \mid x \notin A\}.$$

Theorem

If A is regular, then A^c is regular.

Closure Under Complementation

The *complement* of a language A is

$$A^c = \{x \in \Sigma^* \mid x \notin A\}.$$

Theorem

If A is regular, then A^c is regular.

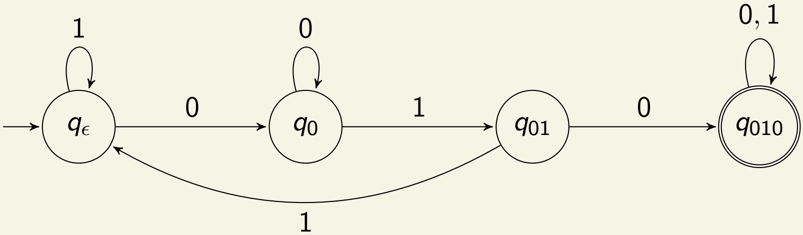
Proof. Let $M = (Q, \Sigma, \delta, q_0, F)$ be a DFA for A. Define a new DFA

$$M' = (Q, \Sigma, \delta, q_0, Q - F)$$

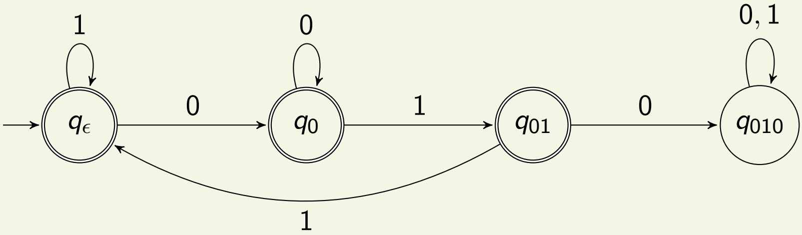
that flips the accepting and nonaccepting states of M. Then $L(M') = L(M)^c = A^c$. $\square$

Example - Closure Under Complementation

$C = \{w \in \{0,1\}^* \mid w \text{ contains } 010 \text{ as a substring}\}.$



$C^c = \{w \in \{0,1\}^* \mid w \text{ does not contain } 010 \text{ as a substring}\}.$



Closure Under Intersection

Theorem

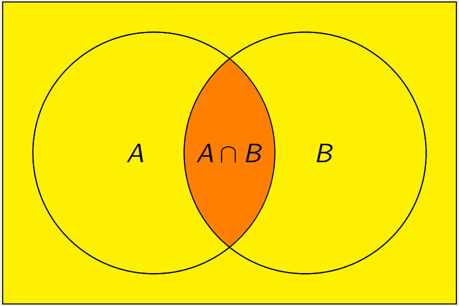
If A and B are regular languages, then $A \cap B$ is regular.

Closure Under Intersection

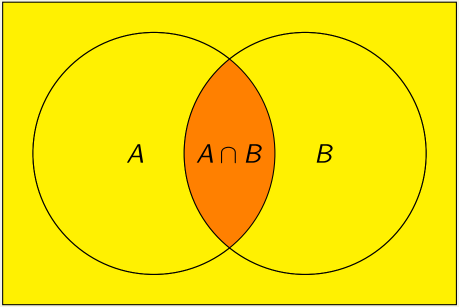
Theorem

If A and B are regular languages, then $A \cap B$ is regular.

Proof. This follows from the previous two theorems. Observe that $A \cap B = (A^c \cup B^c)^c$.



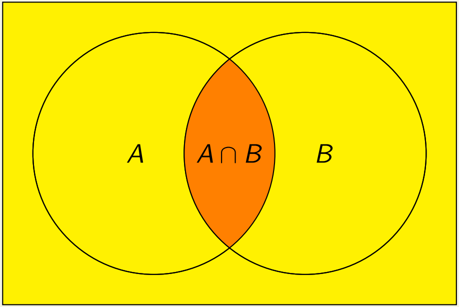
The yellow region is $A^c \cup B^c$.
The complement of the yellow region is $A \cap B$.



The yellow region is $A^c \cup B^c$.
The complement of the yellow region is $A \cap B$.

Formally, we have

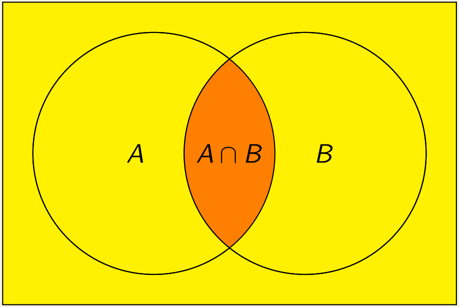
$$(A^c \cup B^c)^c \quad =$$



The yellow region is $A^c \cup B^c$.
The complement of the yellow region is $A \cap B$.

Formally, we have

$$(A^c \cup B^c)^c \quad = \quad (A^c)^c \cap (B^c)^c \quad \text{De Morgan's Law}$$



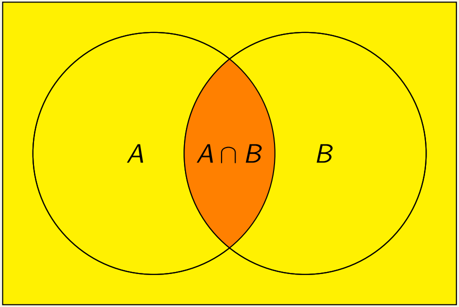
The yellow region is $A^c \cup B^c$.
The complement of the yellow region is $A \cap B$.

Formally, we have

$$\begin{aligned}(A^c \cup B^c)^c &= (A^c)^c \cap (B^c)^c && \text{De Morgan's Law}\\ &= A \cap B && \text{Double Complement Law.}\end{aligned}$$

Now we apply closure properties:

- Since A and B are regular, A^c and B^c is also regular.



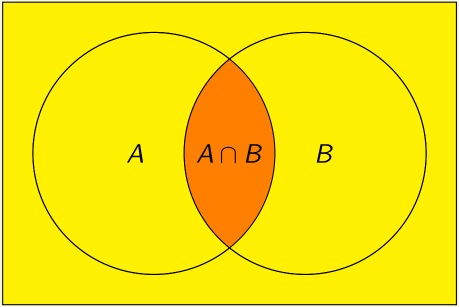
The yellow region is $A^c \cup B^c$.
The complement of the yellow region is $A \cap B$.

Formally, we have

$$\begin{aligned} (A^c \cup B^c)^c &= (A^c)^c \cap (B^c)^c && \text{De Morgan's Law} \\ &= A \cap B && \text{Double Complement Law.} \end{aligned}$$

Now we apply closure properties:

- Since A and B are regular, A^c and B^c is also regular.
- Therefore $A^c \cup B^c$ is regular.



The yellow region is $A^c \cup B^c$.
The complement of the yellow region is $A \cap B$.

Formally, we have

$$\begin{aligned}(A^c \cup B^c)^c &= (A^c)^c \cap (B^c)^c && \text{De Morgan's Law}\\ &= A \cap B && \text{Double Complement Law.}\end{aligned}$$

Now we apply closure properties:

- Since A and B are regular, A^c and B^c is also regular.
- Therefore $A^c \cup B^c$ is regular.
- Finally, $(A^c \cup B^c)^c$ is regular. $\square$

Theorem

If A and B are regular languages, then $A \cap B$ is regular.

We could also prove this by modifying the accepting states in the product construction.

Theorem

If A and B are regular languages, then $A \cap B$ is regular.

We could also prove this by modifying the accepting states in the product construction.

Change the "or" in F to an "and":

$$
\begin{aligned}
F &= \{(p, q) \in Q \mid p \in F_A \text{ or } q \in F_B\} \\
&= F_A \times F_B
\end{aligned}
$$