The regular operations  

For $A$, $B$ languages over same alphabet, define:

$$A \cup B = \{x \mid x \in A \text{ or } x \in B\}$$

$$A \circ B = \{xy \mid x \in A \text{ and } y \in B\}$$

$$A^* = \{x_1x_2 \ldots x_k \mid k \geq 0 \text{ and each } x_i \in A\}$$

These are operations on sets!
Closure of … under …

- $\mathbb{Z}$ under addition.
- Set of even ints under multiplication.
- $\{0\}^*$ under concatenation.

Which of these is true?

A. The set of odd integers is closed under addition.
B. The set of positive integers is closed under subtraction.
C. The set of rational numbers is closed under multiplication.
D. The set of real numbers is closed under division.
E. I don't know.
Complementation

Claim: If $A$ is a regular language over $\{0,1\}^*$, then so is $\overline{A}$
aka "the class of regular languages is closed under complementation"

Proof:
Complementation

Claim: If $A$ is a regular language over $\{0,1\}^*$, then so is $\overline{A}$ aka "the class of regular languages is closed under complementation"

Proof: Let $A$ be a regular language. Then there is a DFA $M=(Q,\Sigma,\delta,q_0,F)$ such that $L(M) = A$. We want to build a DFA whose language is $\overline{A}$. Let

$$M' = (Q,\Sigma,\delta,q_0,\overline{F})$$

Why does $L(M') = A$?
Why closure proofs?

• Stretch the power of the model

• Explore how simple changes in the machine affect the language of the machine

• General technique of proving a new language is regular
The class of regular languages is closed under the union operation.

What are we proving here?

A. For any set A, if A is regular then so is A U A.
B. For any sets A and B, if A U B is regular, then so is A.
C. For two DFAs M1 and M2, M1 U M2 is regular.
D. None of the above.
E. I don't know.
**Theorem:** The class of regular languages over fixed alphabet $\Sigma$ is closed under the union operation.

Proof: Let $A_1$, $A_2$ be any two regular languages over $\Sigma$. \textbf{WTS} that $A_1 \cup A_2$ is regular.

\textbf{Goal:} build a machine that recognizes $A_1 \cup A_2$. 
Goal: build a machine that recognizes $A_1 \cup A_2$.

Strategy: use machines that recognize each of $A_1$, $A_2$.

**HOW?**
Theorem: The class of regular languages over fixed alphabet $\Sigma$ is closed under the union operation.

Proof: Let $A_1$, $A_2$ be any two regular languages over $\Sigma$.
Given $M_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ such that $L(M_1) = A_1$
$M_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$ such that $L(M_2) = A_2$

WTS that $A_1 \cup A_2$ is regular.
Define $M = (?, \Sigma, \delta, ?, ?)$
Theorem: The class of regular languages over fixed alphabet $\Sigma$ is closed under the union operation.

Proof: Let $A_1$, $A_2$ be any two regular languages over $\Sigma$. Given $M_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ such that $L(M_1) = A_1$ and $M_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$ such that $L(M_2) = A_2$.

WTS that $A_1 \cup A_2$ is regular.

Define $M = (Q_1 \times Q_2, \Sigma, \delta, ?, ?)$.

Idea: run in parallel.
Theorem: The class of regular languages over fixed alphabet $\Sigma$ is closed under the union operation.

Proof: Let $A_1$, $A_2$ be any two regular languages over $\Sigma$. Given $M_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ such that $L(M_1) = A_1$ and $M_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$ such that $L(M_2) = A_2$.

WTS that $A_1 \cup A_2$ is regular.

Define $M = (Q_1 \times Q_2, \Sigma, \delta, ?, ?)$

What should be the initial state of $M$?

A. $q_0$
B. $q_1$
C. $q_2$
D. $(q_1, q_2)$
E. I don't know.
Theorem: The class of regular languages over fixed alphabet $\Sigma$ is closed under the union operation.

Proof: Let $A_1, A_2$ be any two regular languages over $\Sigma$.

- M1 = $(Q_1, \Sigma, \delta_1, q_1, F_1)$ such that $L(M_1) = A_1$
- M2 = $(Q_2, \Sigma, \delta_2, q_2, F_2)$ such that $L(M_2) = A_2$

WTS that $A_1 \cup A_2$ is regular.

Define $M = (Q_1 \times Q_2, \Sigma, \delta, ?, ?)$

When $r$ is a state in $M_1$, $s$ is a state in $M_2$, and $x$ is in $\Sigma$, then $\delta( (r,s), x ) =$

A. $(r,s)$
B. $(\delta(r,x), \delta(s,x))$
C. $(\delta_1(r,x), s)$
D. $(\delta_1(r,x), \delta_2(s,x))$
E. I don't know.
Theorem: The class of regular languages over fixed alphabet $\Sigma$ is closed under the union operation.

Proof: Let $A_1$, $A_2$ be any two regular languages over $\Sigma$.

Given $M_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ such that $L(M_1) = A_1$

$M_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$ such that $L(M_2) = A_2$

WTS that $A_1 \cup A_2$ is regular.

Define $M = (Q_1 \times Q_2, \Sigma, \delta, ?, ?)$

The set of accepting states for $M$ is:

A. $F_1 \times F_2$
B. $\{ (r,s) \mid r \text{ is in } F_1 \text{ and } s \text{ is in } F_2 \}$
C. $\{ (r,s) \mid r \text{ is in } F_1 \text{ or } s \text{ is in } F_2 \}$
D. $F_1 \cup F_2$
E. I don't know.
Proof: Let $A_1, A_2$ be any two regular languages over $\Sigma$. Given $M_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ such that $L(M_1) = A_1$ and $M_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$ such that $L(M_2) = A_2$.

**WTS** that $A_1 \cup A_2$ is regular. Define $M = (Q_1 \times Q_2, \Sigma, \delta, (q_1, q_2), \{(r, s) \in Q_1 \times Q_2 \mid r \in F_1 \text{ or } s \in F_2\})$ with $\delta((r, s), x) = (\delta_1(r, x), \delta_2(s, x))$ for each $(r, s) \in Q_1 \times Q_2$ and $x \in \Sigma$.

*Why does $L(M) = A_1 \cup A_2$?*
Intersection

• How would you prove that the class of regular languages is closed under intersection?
• Can you think of more than one proof strategy?

\[ A \cap B = \{ x \mid x \text{ in } A \text{ and } x \text{ in } B \} \]
Payoff

\{ w \mid w \text{ contains neither the substrings } aba \text{ nor } baab \}

Is this a regular set?
Payoff

\{ w \mid w \text{ contains neither the substrings } aba \text{ nor } baab \} \\

Is this a regular set?

A = \{ w \mid w \text{ contains } aba \text{ as a substring} \} \\
B = \{ w \mid w \text{ contains } baab \text{ as a substring} \}

\overline{A \cap B} = \overline{A} \cup \overline{B}
Sample closure proofs for you to try

• The class of regular languages over \{0,1\} is closed under the FlipBits operation, where

\[
\text{FlipBits}(L) = \{ w \mid w \text{ is obtained from some } w' \text{ in } L \text{ by flipping each } 0 \text{ in } w \text{ to } 1, \text{ and each } 1 \text{ to } 0 \}
\]

• The class of regular languages over \{a,b,z\} is closed under the DeleteWordsWithZ operation, where

\[
\text{DeleteWordsWithZ}(L) = \{ w \mid w \text{ is in } L \text{ and } w \text{ doesn't contain } z \}
\]
General proof structure/strategy

**Theorem:** For any $L$ over $\Sigma$, if $L$ is regular then \([ \text{the result of some operation on } L ] \) is also regular.

**Proof:**
1. **Given.** Name the variables for sets, machines assumed to exist.
2. **Goal.** State what you want to show.
3. **Construction.** Use objects previously defined + new tools working towards goal. *Explain how to construct new machine.*
4. **Strategy.** Explain why construction recognizes desired language.
5. **Recap.** Your construction shows closure under operation.
The regular operations

For A, B languages over same alphabet, define:

✔

\[ A \cup B = \{ x | x \in A \text{ or } x \in B \} \]

\[ A \circ B = \{ xy \mid x \in A \text{ and } y \in B \} \]

\[ A^* = \{ x_1x_2 \ldots x_k \mid k \geq 0 \text{ and each } x_i \in A \} \]

How can we prove that the concatenation of two regular languages is a regular language?