Today's learning goals  Sipser Ch 1.1

• General Properties of Regular Languages
• Operations on languages
• Closure properties
Recall terminology

- **Alphabet**: nonempty finite set of **symbols**
- **String** over an alphabet: finite sequence of symbols
- **Language** over an alphabet: some set of strings

- **DFA** over an alphabet: deterministic finite automaton
  - Input: finite string over a fixed alphabet
  - Output: "accept" or "reject"
  - \( L(M) = \{ w \mid M \text{ accepts } w \} \)
- **Regular language**
  language that is \( L(M) \) for some DFA \( M \)
The regular operations \textbf{Sipser Def 1.23 p. 44}

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

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

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

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

These are operations on sets of \textit{strings}!
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, then so is its complement $A^c$.

Same as:
If $A=\mathcal{L}(M)$ for some DFA $M$, then $A^c=\mathcal{L}(M')$ for some DFA $M'$

Proof Strategy: Show that any DFA $M$ can be transformed into a DFA $M'$ such that $\mathcal{L}(M') = \overline{\mathcal{L}(M)}$
Complementation

Claim: If $A$ is a regular language, then so is $\overline{A}$

Proof:

1) Assume $A$ is regular
2) By definition $A = L(M)$ for some DFA $M = (Q, \Sigma, \delta, s, F)$
3) Let $M' = (Q, \Sigma, \delta, s, F')$
4) Claim: $\overline{A} = L(M')$
5) Therefore $\overline{A}$ is also regular

How would you define $F'$?

A) $F' = Q - \{s\}$
B) $F' = F - \{s\}$
C) $F' = Q - F$
D) $F' = \{\}$
Complementation (Proof details)

Claim: Let \( M=(Q, \Sigma, \delta, s, F) \) and \( M'=(Q, \Sigma, \delta, s, F) \) be DFAs. Then \( L(M') = L(M) \)

Proof:

- \((w \in L(M')) \rightarrow (w \in L(M))\)
  1) Assume \( w \) is in \( L(M') \)
  2) By definition of \( L(M') \), \( \delta^*(s,w) \) is in \( F \)
  3) So, \( \delta^*(s,w) \) is not in \( F \), and \( w \) is not in \( L(M) \)
  4) Therefore, \( w \) is in \( L(M) \)

- \((w \in L(M)) \rightarrow (w \in L(M'))\): similar proof
**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.

**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 $L(M_1) = A_1$ and $M_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$ such that $L(M_2) = A_2$ and 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 $L(M_2) = A_2$ and WTS that $A_1 \cup A_2$ is regular.

Define $M = (Q_1 \times Q_2, \Sigma, \delta, ?, ?)$.
The class of regular languages over a 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 $M = (Q_1 \times Q_2, \Sigma, \delta, ?, ?)$ where

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 $\Sigma$. 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 $L(M_2) = A_2$ and WTS that $A_1 \cup A_2$ is regular.

Define $M = (Q_1 \times Q_2, \Sigma, \delta, (q_1, q_2), ?)$

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 $L(M_1) = A_1$, 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 $M = (Q_1 \times Q_2, \Sigma, \delta, (q_1, q_2), F_1 \times F_2)$.

The set of accepting states for $M$ is

A. $F_1 \times F_2$
B. $\{ (r,s) | r \text{ is in } F_1 \text{ and } s \text{ is in } F_2 \}$
C. $\{ (r,s) | r \text{ is in } F_1 \text{ or } s \text{ is in } F_2 \}$
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 $\Sigma$.

Claim that $L(M) = A_1 \cup A_2$. Proof...
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 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} \}

\bar{A} \cap \bar{B} = \overline{A \cup B}
Sample closure proofs

• The class of regular languages over \{0,1\} is closed under the FlipBits operation, where
  \[ \text{FlipBits}(L) = \{ w \mid w \text{ is obtained from some } w' \text{ in } L \text{ by flipping each 0 in } w \text{ to 1, and each 1 to 0} \} \]

• The class of regular languages of \{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 [the result of some operation on $L$] is also regular.

**Proof:**

*Given* name variables for sets, machines assumed to exist.

*WTS* state goal and outline plan.

*Construction* using objects previously defined + new tools working towards goal. Give formal definition and explain.

*Correctness* prove that construction works.

*Conclusion* recap what you've proved.
For next time

Start working on

   HW2   (discussion: Today, Due: next Wednesday)

Next Time:

*Class of regular languages is also closed under concatenation and Kleene star, but harder to prove*