Regular operations

Sipser 1.1 (pages 44 – 47)

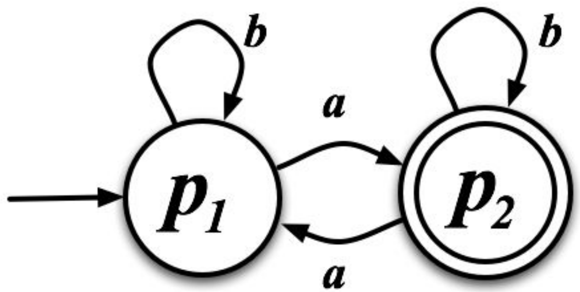
Building languages

- If L is a language, then its **complement** is $L' = \{w \mid w \notin L\}$

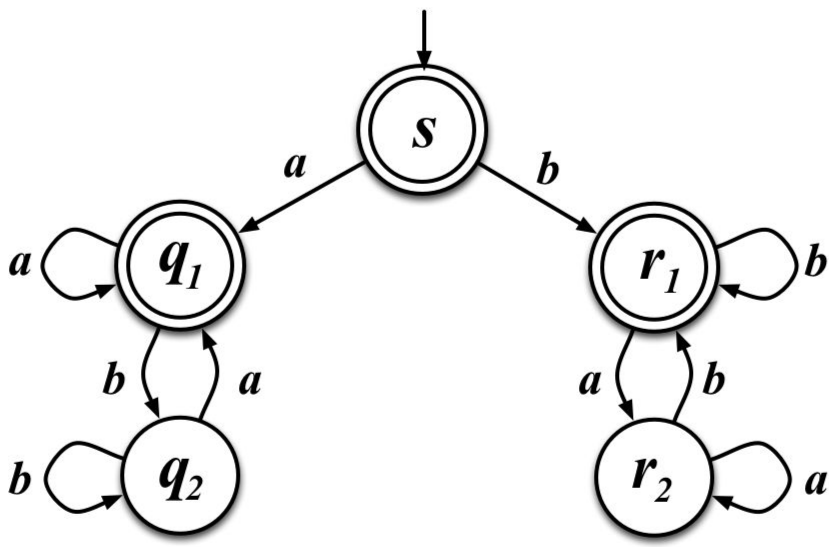
- Let $A = \{w \mid w$ is a string of as and bs containing an odd number of as$\}$. What is A'?
- Let $B = \{w \mid w$ is a string of over $\{a,b\}$ that starts and ends with the same symbol}. What is B'?

Which complements are regular?

- $A = \{w \mid w$ is a string of as and bs containing an odd number of as$\}$
- What about A'?



- $B = \{w \mid w$ is a string over $\{a,b\}$ that starts and ends with the same symbol}
- What about B'?



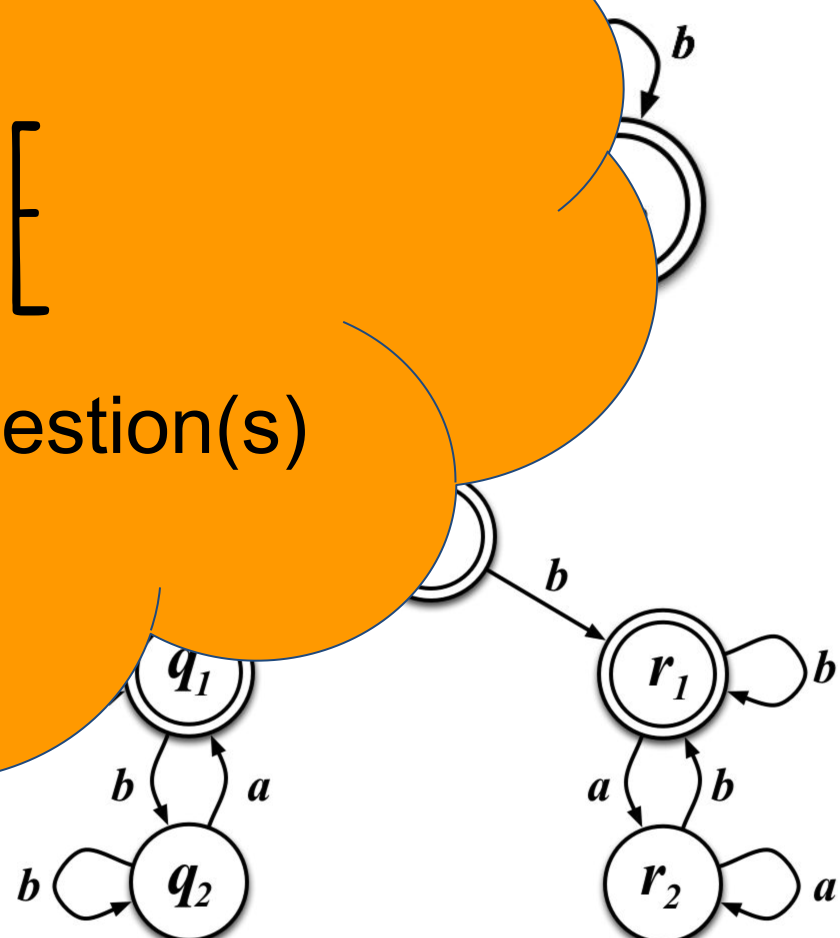
Which complements are

Please

PAUSE

to answer the question(s) below!

- $A = \{w$
  and
  nu
- W
- B
  $\{a,b\}$
  with the
- What about B



Our first theorem

- Theorem: The class of regular languages is *closed* under the complement operator
- How do we prove it?

Our first proof

- Theorem: The class of regular languages is *closed* under the *complement* operation
- Proof:

Let A be a regular language. By definition, there exists a finite automaton

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

that recognizes $A = L(M)$. Since

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

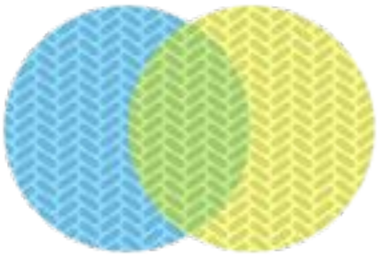
is a finite automaton that recognizes $L(N) = A'$, A' is a regular language.

Union and Intersection

- Let A and B be languages

- Define their ***union***

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



- Define their ***intersection***

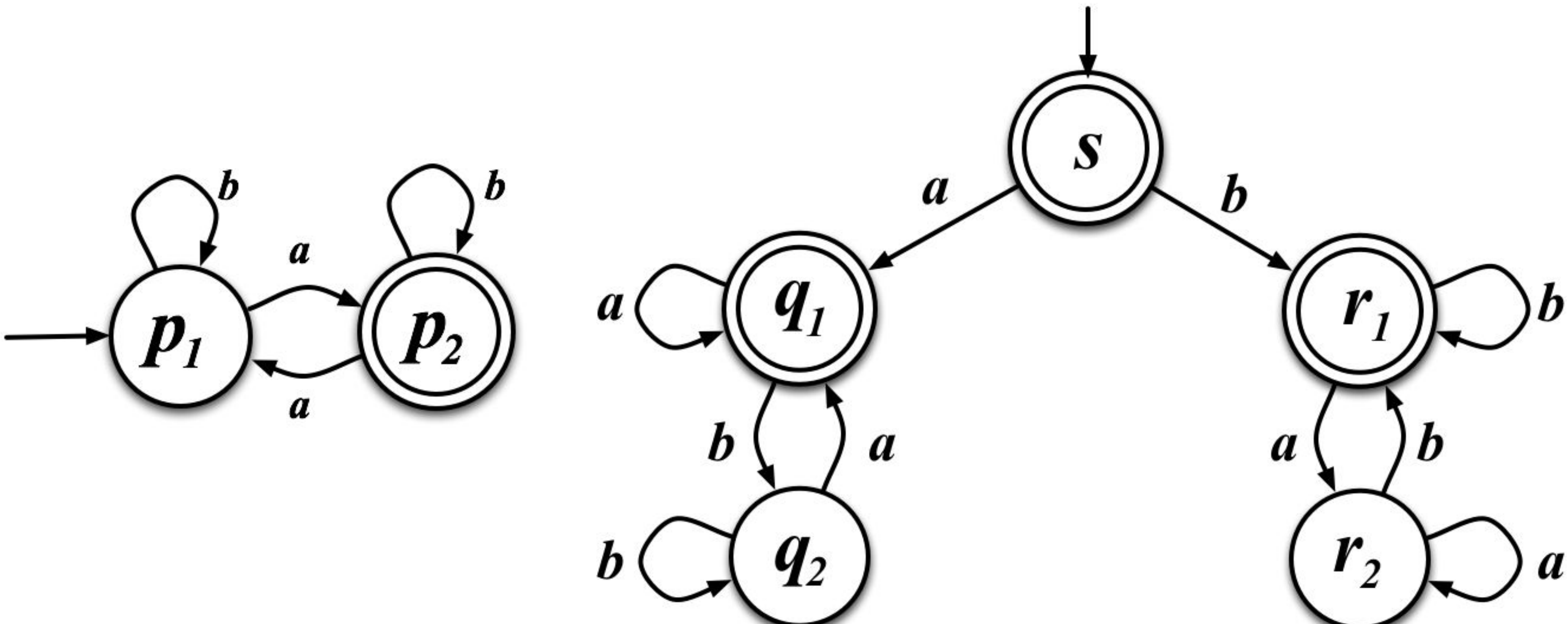
  $A \cap B = \{ w \mid w \in A \text{ and } w \in B \}$

Example: union

- $A = \{w \mid w$ is a string over $\{a,b\}$ containing an odd number of as$\}$
- $B = \{w \mid w$ is a string over $\{a,b\}$ that starts and ends with the same symbol}

- What is $A \cup B$?
- Is $A \cup B$ regular?

Can we build a recognizer for union from previous machines?



Now do it in general… our second theorem

- Theorem: The class of regular languages is *closed* under the *union* operation

The proof

- Theorem: The class of regular languages is *closed* under the *union* operation
- Let A and B be regular languages. By definition, there exists finite automata $M_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ and $M_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$ that recognize $A = L(M_1)$ and $B = L(M_2)$, respectively.

  What's left? Build a finite automaton M that recognizes $L(M) = A \cup B$.

What about intersection?

- Theorem: The class of regular languages is *closed* under the *intersection* operation

Proof formalities

We construct a new DFA

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

$Q = Q_1 \times Q_2$

$\delta(\underbrace{(\overset{\in Q_1}{r}, \overset{\in Q_2}{p})}_{\in Q}, \underbrace{\sigma}_{\in \Sigma}) = \underbrace{(\overbrace{\delta_1(r,\sigma)}^{\in Q_1}, \overbrace{\delta_2(p,\sigma)}^{\in Q_2})}_{\in Q}$

$q_0 = (q_1, q_2)$

$F = \{ (r,p) \mid r \in F_1$ ~~or~~ and $p \in F_2 \}$

Since $L(M) = A \cap B$, then $A \cap B$ is a regular language. Thus, the class of reg. lang is closed under ~~union~~ intersection.

Cartesian product $A \times B$

$\{(a,b) \mid a \in A \text{ and } b \in B\}$

$\delta : Q \times \Sigma \to Q$

public State delta (State q, Symbol σ)