Overview

⇒ Deduction
• Construction
• Contradiction
• Induction
• Closure Properties of Regular Languages

Deduction

• Not in book
• From a set of facts, deduce something that must be true
  L is a regular language
  So there must exist a DFA $M$ that recognizes it
  $w$ is a string in the alphabet of $\Sigma$
  So there exists an $n \geq 0$ such that $w = w_1 \ldots w_n$ and $w_i \in \Sigma$
• Show something is equivalent to a different problem
  Prove that regular languages are closed under complementation
  Let $L$ be any regular language
  Sufficient to show that $\overline{L}$ is regular
  Sufficient to show there exists a DFA that recognizes $\overline{L}$
Overview

- Deduction
  \[ \Rightarrow \] Construction
- Contradiction
- Induction
- Closure Properties of Regular Languages
Proof by Construction

- Often asked to prove that a particular type of object exists
  - Question usually along the lines of
    Prove that there exists ...
    Prove that for each \( n > 2 \) there exists ...
- Proof technique:
  - Prove by coming up with an example
  - Or demonstrating how to construct the object
- Sometimes disguised
  - Show that \( a^*b^* \) is regular
  - \( a^*b^* \) is regular iff there a DFA that recognizes \( a^*b^* \)
  - Prove that regular languages are closed under complementation
  - Let \( L \) be any regular language
    Sufficient to show that \( \bar{L} \) is regular
    Sufficient to show there exists a DFA that recognizes \( \bar{L} \)

Regular Languages Closed under Complementation

Let \( L \) be any regular language
Sufficient to show that \( \bar{L} \) is regular
Sufficient to show there exists a DFA that recognizes \( \bar{L} \)
Since \( L \) is regular, there exists a DFA that recognizes \( \bar{L} \)
Let \( M = (Q, \Sigma, \delta, q_0, F) \)
Construct DFA \( M' = (Q, \Sigma, \delta, q_0, F') \) where \( F' = Q - F \)
**Claim:** \( L(M') = \bar{L} \)
Let \( w \in L \)
So \( M' \) accepts \( w \)
Let \( q \) be the state that \( M' \) is in at the end of processing \( w \)
So \( q \in F' \).
So when \( M' \) processes \( w \), at the end of \( w \), it will also be in \( q \).
Since \( q \notin F' \), \( M' \) does not accept \( w \)
So \( w \notin L(M') \)
Similarly, if \( w \in L(M') \) then \( w \notin L(M') \)
So \( L(M') = \bar{L} \)
How Much Detail?

- How much detail is needed?
  - Need to convince me that you know how to do the proof
  - Should be clear what you need to prove (2nd and 3rd line)
- Do you need to show construction does as is intended?
  - Sometimes you can handwave
  - Or question might specify to just give the construction
- Notice the word ‘let’, as in ‘let $x$ be ...’
  - Used when it is obvious that such an object exists, but you need a name
to refer to any object of that type
- Is the last line obvious enough?
  - Could add in:
    So, if $w \in L$, then $w \notin L(M')$, and if $w \in L(M')$ then $w \notin L$
    So $L(M') = \overline{M}$

Overview

- Deduction
- Construction
  $\Rightarrow$ Contradiction
- Induction
- Closure Properties of Regular Languages
Proof by Contradiction

- If you have to prove $X$, assume that $X$ is false, and show that you get a contradiction.
- Sometimes this is easier than trying to directly prove $X$.

Prove $a^n b^n$ is not regular

Let $A = \{a^i b^i | i \geq 0\}$. Assume $A$ is regular.

So there exists DFA $M$ such that $L(M) = A$.

Let $n$ be the number of states that $M$ has.

Let $s = a^{n+1} b^{n+1}$.

Obviously $s \in L(M)$.

Let $r_0, \ldots , r_{2n+2}$ be the state sequence for accepting $s$.

Now consider the state sequence from $r_0$ to $r_n$ in which each $r_k$ on input $a$ transitions to $r_{k+1}$ for $k \leq n$ (i.e., $\delta(r_k, a) = r_{k+1}$).

Since $M$ just has $n$ states, there must be at least one duplicate.

Let $r_i$ and $r_{i+j}$ be duplicate states, with $j > 0$, so $r_i = r_{i+j}$.

Since $r_{i+j+1} = \delta(r_{i+j}, a)$ then $r_{i+2j+1} = \delta(r_i, a)$.

So, $r_0, r_{i+j+1}, r_{2n+2}$ is a valid state sequence and accepts $a^{n+1} b^{n+1}$.

So, we can cut out the intervening $j$ states and still have an accepting state sequence.

So $a^{n+1-2j} b^{n+1} \in L(M)$.

Contradiction, so $A$ is not regular.
Overview

- Deduction
- Construction
- Contradiction
- Induction
- Closure Properties of Regular Languages

Induction

- Prove that all Xs have a certain property where all Xs can be categorized in terms of some property $P$ based on the natural numbers, such as:
  - number of states in a DFA
  - number of nodes in a graph
  - some variable that is restricted to the natural numbers

- Break problem into a base case and an induction step
  - Base case: prove that $P(1)$ is true
  - Induction step: prove that if $P(i)$ is true for $i \geq 1$ then so is $P(i + 1)$
  - After proven both parts, you know that $P(i)$ must be true for $i \geq 1$
    - Proved that $P(1)$ is true. Since $P(1)$ is true, so its $P(2)$. Since $P(2)$ is true, so is $P(3)$. Since $P(3)$ is true, so $P(4)$ etc
Continued

- Many variations of this:
  - Starting at a number other than 1
  - Needing to prove two base cases $P(1)$ and $P(2)$
  - Needing to assume $P(j)$ is true for all $j \leq i$ in order to prove $P(i + 1)$

---

Prove that $\sum_{i=1}^{n} i = n(n + 1)/2$ for $n \geq 1$

**Proof by Induction:**

**Base case:** prove for $n = 1$

\[
LHS = \sum_{i=1}^{n} i = 1 \quad (1)
\]

\[
RHS = n(n + 1)/2 = 1 * 2/2 = 1 \quad (3)
\]

Since LHS = RHS, true for base case
Continued

**Induction step:** Assume it is true for \( n \), prove it is true for \( n + 1 \)

So assuming \( \sum_{i=1}^{n} i = n(n + 1)/2 \)

Need to prove that \( \sum_{i=1}^{n+1} i = (n + 1)(n + 2)/2 \)

\[
LHS = \sum_{i=1}^{n+1} i
= \sum_{i=1}^{n} i + (n + 1)
= n(n + 1)/2 + (n + 1)
= (n(n + 1) + 2(n + 1))/2
= (n^2 + 3n + 2)/2
\]

\[
RHS = (n + 1)(n + 2)/2
= (n^2 + 3n + 2)/2
\]

Since LHS = RHS, true for \( n + 1 \).

---

**Overview**

- Deduction
- Construction
- Contradiction
- Induction

\( \Rightarrow \) Closure Properties of Regular Languages
Closed under Union

**Definition of Union:** $A \cup B = \{ x | x \in A \text{ or } x \in B \}$

- Class of regular languages is closed under union
  - If $A_1$ and $A_2$ are regular languages (each accepted by a FA), then there is a FA $M$ that accepts $A_1 \cup A_2$
- Proof by construction:
  - Since $A_1$ and $A_2$ are regular languages, there exists FA $M_1$ and $M_2$ such that $L(M_1) = A_1$ and $L(M_2) = A_2$
  - Let’s construct $M$ so that $M$ accepts $w$ if and only if one of $M_1$ and $M_2$ would accept it
  - Have $M$ simulate the operations of $M_1$ and $M_2$

Proof Idea (continued)

- **Approach 1:**
  - We could have $M$ first simulate $M_1$ and then simulate $M_2$
  - This won’t work because in simulating $M_1$, we will have used up the input with $w$ on it, and we won’t be able to get it back
- **Approach 2:**
  - Simulate both $M_1$ and $M_2$ at the same time!
  - Have the states of $M$ be the product of states of $M_1$ and $M_2$
  - Have the transition function for $M$ on $w_i$ transition to the new state depending on what $M_1$ and $M_2$ would have individually did
  - Have accepting states of $M$ be any state with an accepting state from $M_1$ or from $M_2$
Example

• Let $\Sigma = \{0, 1\}$
• Let $A_1$ be strings in which the number of 0’s is divisible by 3
  - Draw a state diagram for $M_1$ with 3 states: $a$, $b$, $c$
  - Draw the 3 states horizontally

• Let $A_2$ be strings that have an even number of 1s
  - Draw a state diagram for $M_1$ with 2 states: $d$, $e$
  - Draw the two states vertically

Example Continued

• What are the states of $M$ (draw them in a 3x2 array)

• What are the transitions of $M$?
  - Take each state, like $ad$, and each input, like 0, and ask:
    where does $M_1$ transition to from $a$ on 0?
    where does $M_2$ transition to from $d$ on 0?
Formal Proof

Let $M_1$ recognize $A_1$ where $M_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$
Let $M_2$ recognize $A_2$ where $M_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$

- Construct $M$ to recognize $A_1 \cup A_2$ where $M = (Q, \Sigma, \delta, q_0, F)$
  - $Q$:
  - $\delta$:
  - $q_0$:
  - $F$:

Other Operations

- Are regular languages closed under
  - Intersection? $A \cap B = \{x | x \in A \text{ and } x \in B\}$
  - Complementation? $\overline{A} = \{x | x \notin A\}$
  - Concatenation? $A \circ B = \{xy | x \in A \text{ and } y \in B\}$
  - Star? $A^* = \{x_1x_2...x_k | k \geq 0 \text{ and each } x_i \in A\}$