Find the regular expressions for the following languages (if possible)

1. \( L_1 = \{0^m1^n | m, n \geq 0\} \)

2. \( L_2 = \{0^n1^n | n \geq 0\} \)

3. \( L_3 = L_1 \cup L_2 \)

4. \( L_4 = L_1 \cap L_2 \)
Pre-lecture brain teaser

Find the regular expressions for the following languages (if possible)

1. \( L_1 = \{0^m1^n | m, n \geq 0\} \)

2. \( L_2 = \{0^n1^n | n \geq 0\} \)

3. \( L_3 = L_1 \cup L_2 \)

4. \( L_4 = L_1 \cap L_2 \)
We have a language \( L = \{0^n1^n | n \geq 0\} \)
Prove that \( L \) is non-regular.
Proving non-regularity: Methods

- **Pumping lemma.** We will not cover it but it is *sometimes* an easier proof technique to apply, but not as general as the fooling set technique.

- **Closure** properties. Use existing non-regular languages and regular languages to prove that some new language is non-regular.

- **Fooling sets**- Method of distinguishing suffixes. To prove that $L$ is non-regular find an infinite fooling set.
Not all languages are regular
Theorem
Languages accepted by DFAs, NFAs, and regular expressions are the same.

Question: Is every language a regular language? No.
Theorem
Languages accepted by DFAs, NFAs, and regular expressions are the same.

Question: Is every language a regular language? No.

- Each DFA $M$ can be represented as a string over a finite alphabet $\Sigma$ by appropriate encoding
- Hence number of regular languages is countably infinite
- Number of languages is uncountably infinite
- Hence there must be a non-regular language!
A Simple and Canonical Non-regular Language

$L = \{0^n1^n \mid n \geq 0\} = \{\epsilon, 01, 0011, 000111, \ldots\}$

Theorem

$L$ is not regular.

Question:
Proof?

Intuition:
Any program to recognize $L$ seems to require counting the number of zeros in the input, which cannot be done with fixed memory.

How do we formalize intuition and come up with a formal proof?
Let $L = \{0^n1^n \mid n \geq 0\} = \{\epsilon, 01, 0011, 000111, \ldots\}$.

**Theorem**
$L$ is not regular.
A Simple and Canonical Non-regular Language

$L = \{0^n1^n \mid n \geq 0\} = \{\epsilon, 01, 0011, 000111, \ldots\}$

**Theorem**
$L$ is not regular.

**Question:** Proof?
A Simple and Canonical Non-regular Language

\[ L = \{0^n1^n \mid n \geq 0\} = \{\epsilon, 01, 0011, 000111, \cdots\} \]

**Theorem**

\( L \) is not regular.

**Question:** Proof?

**Intuition:** Any program to recognize \( L \) seems to require counting number of zeros in input which cannot be done with fixed memory.
A Simple and Canonical Non-regular Language

$L = \{0^n1^n \mid n \geq 0\} = \{\epsilon, 01, 0011, 000111, \cdots\}$

**Theorem**

$L$ is not regular.

**Question:** Proof?

**Intuition:** Any program to recognize $L$ seems to require counting number of zeros in input which cannot be done with fixed memory.

How do we formalize intuition and come up with a formal proof?
Proof by contradiction

- Suppose $L$ is regular. Then there is a DFA $M$ such that $L(M) = L$.
- Let $M = (Q, \{0, 1\}, \delta, s, A)$ where $|Q|$ is finite.
Proof by contradiction

• Suppose \( L \) is regular. Then there is a DFA \( M \) such that \( L(M) = L \).
• Let \( M = (Q, \{0, 1\}, \delta, s, A) \) where \(|Q|\) is finite.
Proof by Contradiction

- Suppose $L$ is regular. Then there is a DFA $M$ such that $L(M) = L$.
- Let $M = (Q, \{0, 1\}, \delta, s, A)$ where $|Q| = n$. 
Proof by Contradiction

• Suppose $L$ is regular. Then there is a DFA $M$ such that $L(M) = L$.
• Let $M = (Q, \{0, 1\}, \delta, s, A)$ where $|Q| = n$.

Consider strings $\epsilon, 0, 00, 000, \cdots, 0^n$ total of $n + 1$ strings.
Proof by Contradiction

• Suppose \( L \) is regular. Then there is a DFA \( M \) such that 
  \( L(M) = L \).
• Let \( M = (Q, \{0, 1\}, \delta, s, A) \) where \( |Q| = n \).

Consider strings \( \epsilon, 0, 00, 000, \cdots, 0^n \) total of \( n + 1 \) strings.

What states does \( M \) reach on the above strings? Let 
\( q_i = \delta^*(s, 0^i) \).

By pigeon hole principle \( q_i = q_j \) for some \( 0 \leq i < j \leq n \).
That is, \( M \) is in the same state after reading \( 0^i \) and \( 0^j \) where 
\( i \neq j \).
Proof by Contradiction

• Suppose $L$ is regular. Then there is a DFA $M$ such that $L(M) = L$.
• Let $M = (Q, \{0, 1\}, \delta, s, A)$ where $|Q| = n$.

Consider strings $\epsilon, 0, 00, 000, \cdots, 0^n$ total of $n + 1$ strings.

What states does $M$ reach on the above strings? Let $q_i = \delta^*(s, 0^i)$.

By pigeon hole principle $q_i = q_j$ for some $0 \leq i < j \leq n$. That is, $M$ is in the same state after reading $0^i$ and $0^j$ where $i \neq j$.

$M$ should accept $0^i1^i$ but then it will also accept $0^j1^i$ where $i \neq j$. 
Proof by Contradiction

- Suppose $L$ is regular. Then there is a DFA $M$ such that $L(M) = L$.
- Let $M = (Q, \{0, 1\}, \delta, s, A)$ where $|Q| = n$.

Consider strings $\epsilon, 0, 00, 000, \cdots, 0^n$ total of $n + 1$ strings.

What states does $M$ reach on the above strings? Let $q_i = \delta^*(s, 0^i)$.

By pigeon hole principle $q_i = q_j$ for some $0 \leq i < j \leq n$. That is, $M$ is in the same state after reading $0^i$ and $0^j$ where $i \neq j$.

$M$ should accept $0^i1^i$ but then it will also accept $0^j1^i$ where $i \neq j$. This contradicts the fact that $M$ accepts $L$. Thus, there is no DFA for $L$. 
When two states are equivalent?
States that cannot be combined?

We concluded that because each $0^i$ prefix has a unique state. Are there states that aren’t unique? Can states be combined?
**Equivalence between states**

**Definition**

$M = (Q, \Sigma, \delta, s, A)$: DFA.

Two states $p, q \in Q$ are **equivalent** if for all strings $w \in \Sigma^*$, we have that

$$\delta^*(p, w) \in A \iff \delta^*(q, w) \in A.$$  

One can merge any two states that are equivalent into a single state.
Distinguishing between states

**Definition**

\[ M = (Q, \Sigma, \delta, s, A) : \text{DFA.} \]

Two states \( p, q \in Q \) are **distinguishable** if there exists a string \( w \in \Sigma^* \), such that

\[ \delta^*(p, w) \in A \quad \text{and} \quad \delta^*(q, w) \notin A. \]

or

\[ \delta^*(p, w) \notin A \quad \text{and} \quad \delta^*(q, w) \in A. \]
Distinguishable prefixes

\[ M = (Q, \Sigma, \delta, s, A): \text{DFA} \]

**Idea:** Every string \( w \in \Sigma^* \) defines a state \( \nabla w = \delta^*(s, w) \).
Distinguishable prefixes

\[ M = (Q, \Sigma, \delta, s, A): \text{DFA} \]

**Idea:** Every string \( w \in \Sigma^* \) defines a state \( \nabla w = \delta^*(s, w) \).

**Definition**
Two strings \( u, w \in \Sigma^* \) are **distinguishable** for \( M \) (or \( L(M) \)) if \( \nabla u \) and \( \nabla w \) are distinguishable.

**Definition (Direct restatement)**
Two prefixes \( u, w \in \Sigma^* \) are **distinguishable** for a language \( L \) if there exists a string \( x \), such that \( ux \in L \) and \( wx \notin L \) (or \( ux \notin L \) and \( wx \in L \)).

![DFA Diagram]

- Start state: \( q_0 \)
- States: \( q_0, q_1, q_2, q_3, q_4 \)
- Transitions:
  - \( q_0 \) to \( q_1 \) on 0 and 0
  - \( q_0 \) to \( q_2 \) on 1
  - \( q_1 \) to \( q_3 \) on 0
  - \( q_1 \) to \( q_4 \) on 1
  - \( q_2 \) to \( q_0 \) on 0
  - \( q_2 \) to \( q_3 \) on 1
  - \( q_3 \) to \( q_2 \) on 0
  - \( q_3 \) to \( q_4 \) on 1
  - \( q_4 \) loop on 0,1
Distinguishable means different states

**Lemma**

$L$: regular language.

$M = (Q, \Sigma, \delta, s, A)$: DFA for $L$.

If $x, y \in \Sigma^*$ are distinguishable, then $\nabla x \neq \nabla y$.

Reminder: $\nabla x = \delta^*(s, x) \in Q$ and $\nabla y = \delta^*(s, y) \in Q$
Proof by a figure

Possible

\[ \delta^*(s, x) \xrightarrow{w} \delta^*(s, xw) \]
\[ \delta^*(s, y) \xrightarrow{w} \delta^*(s, yw) \]

Not possible

\[ \delta^*(s, x) = \delta^*(s, y) \]
\[ \delta^*(s, xw) \xrightarrow{w} \delta^*(s, yw) \]
Review questions...

• Are \( \nabla 0^i \) and \( \nabla 0^j \) are distinguishable for the language \( \{0^n 1^n \mid n \geq 0\} \).
Review questions...

• Are \( \nabla 0^i \) and \( \nabla 0^j \) are distinguishable for the language \( \{0^n1^n \mid n \geq 0\} \).

• Let \( L \) be a regular language, and let \( w_1, \ldots, w_k \) be strings that are all pairwise distinguishable for \( L \). How many states must the DFA for \( L \) have?

• Prove that \( \{0^n1^n \mid n \geq 0\} \) is not regular.
Review questions...

• Are \( \nabla 0^i \) and \( \nabla 0^j \) are distinguishable for the language \( \{0^n1^n \mid n \geq 0\} \).

• Let \( L \) be a regular language, and let \( w_1, \ldots, w_k \) be strings that are all pairwise distinguishable for \( L \). How many states must the DFA for \( L \) have?

• Prove that \( \{0^n1^n \mid n \geq 0\} \) is not regular.
Fooling sets: Proving non-regularity
Fooling Sets

Definition
For a language $L$ over $\Sigma$ a set of strings $F$ (could be infinite) is a fooling set or distinguishing set for $L$ if every two distinct strings $x, y \in F$ are distinguishable.
Definition
For a language \( L \) over \( \Sigma \) a set of strings \( F \) (could be infinite) is a fooling set or distinguishing set for \( L \) if every two distinct strings \( x, y \in F \) are distinguishable.

Example: \( F = \{0^i \mid i \geq 0\} \) is a fooling set for the language \( L = \{0^n1^n \mid n \geq 0\} \).
**Fooling Sets**

**Definition**
For a language $L$ over $\Sigma$ a set of strings $F$ (could be infinite) is a fooled set or distinguishing set for $L$ if every two distinct strings $x, y \in F$ are distinguishable.

**Example:** $F = \{0^i \mid i \geq 0\}$ is a fooling set for the language $L = \{0^n1^n \mid n \geq 0\}$.

**Theorem**
Suppose $F$ is a fooling set for $L$. If $F$ is finite then there is no DFA $M$ that accepts $L$ with less than $|F|$ states.
Recall

Already proved the following lemma:

**Lemma**

$L$: regular language.

$M = (Q, \Sigma, \delta, s, A)$: **DFA** for $L$.

*If* $x, y \in \Sigma^*$ *are distinguishable, then* $\nabla x \neq \nabla y$.

Reminder: $\nabla x = \delta^*(s, x)$. 
Theorem (Reworded.)
$L$: A language

$F$: a fooling set for $L$.

If $F$ is finite then any DFA $M$ that accepts $L$ has at least $|F|$ states.

Proof.
Let $F = \{w_1, w_2, \ldots, w_m\}$ be the fooling set.

Let $M = (Q, \Sigma, \delta, s, A)$ be any DFA that accepts $L$.

Let $q_i = \nabla w_i = \delta^*(s, x_i)$.

By lemma $q_i \neq q_j$ for all $i \neq j$.

As such, $|Q| \geq |\{q_1, \ldots, q_m\}| = |\{w_1, \ldots, w_m\}| = |A|$. \qed
Corollary
If L has an infinite fooling set F then L is not regular.

Proof.
Let $w_1, w_2, \ldots \subseteq F$ be an infinite sequence of strings such that every pair of them are distinguishable.

Assume for contradiction that $\exists M$ a DFA for $L$. 
Corollary
If $L$ has an infinite fooling set $F$ then $L$ is not regular.

Proof.
Let $w_1, w_2, \ldots \subseteq F$ be an infinite sequence of strings such that every pair of them are distinguishable.

Assume for contradiction that $\exists M$ a DFA for $L$.

Let $F_i = \{w_1, \ldots, w_i\}$.

By theorem, $\# \text{ states of } M \geq |F_i| = i$, for all $i$.

As such, number of states in $M$ is infinite.
Corollary
If $L$ has an infinite fooling set $F$ then $L$ is not regular.

Proof.
Let $w_1, w_2, \ldots \subseteq F$ be an infinite sequence of strings such that every pair of them are distinguishable.

Assume for contradiction that $\exists M$ a DFA for $L$.

Let $F_i = \{w_1, \ldots, w_i\}$.

By theorem, $\# \text{states of } M \geq |F_i| = i$, for all $i$.

As such, number of states in $M$ is infinite.

Contradiction: DFA = deterministic finite automata. But $M$ not finite. 

Examples

- \{0^n1^n \mid n \geq 0\}
Examples

• $\{0^n1^n \mid n \geq 0\}$

• \{bitstrings with equal number of 0s and 1s\}
Examples

• $\{0^n1^n \mid n \geq 0\}$

• \{bitstrings with equal number of 0s and 1s\}

• $\{0^k1^\ell \mid k \neq \ell\}$
$L = \{\text{strings of properly matched open and closing parentheses}\}$
Examples

$L = \{\text{palindromes over the binary alphabet} \Sigma = \{0, 1\}\}$

A palindrome is a string that is equal to its reversal, e.g. 10001 or 0110.
Closure properties: Proving non-regularity
Non-regularity via closure properties

\[ H = \{ \text{bitstrings with equal number of 0s and 1s} \} \]

\[ H' = \{ 0^k1^k \mid k \geq 0 \} \]

Suppose we have already shown that \( L' \) is non-regular. Can we show that \( L \) is non-regular without using the fooling set argument from scratch?

Suppose \( H \) is regular. Then since \( L(0^*1^*) \) is regular, and regular languages are closed under intersection, \( H' \) also would be regular. But we know \( H' \) is not regular, a contradiction.
Non-regularity via closure properties

\[ H = \{ \text{bitstrings with equal number of 0s and 1s} \} \]

\[ H' = \{ 0^k1^k \mid k \geq 0 \} \]

Suppose we have already shown that \( L' \) is non-regular. Can we show that \( L \) is non-regular without using the fooling set argument from scratch?

\[ H' = H \cap L(0^*1^*) \]

**Claim:** The above and the fact that \( L' \) is non-regular implies \( L \) is non-regular. Why?
Non-regularity via closure properties

\[ H = \{ \text{bitstrings with equal number of 0s and 1s} \} \]

\[ H' = \{ 0^k1^k \mid k \geq 0 \} \]

Suppose we have already shown that \( L' \) is non-regular. Can we show that \( L \) is non-regular without using the fooling set argument from scratch?

\[ H' = H \cap L(0^*1^*) \]

**Claim:** The above and the fact that \( L' \) is non-regular implies \( L \) is non-regular. Why?

Suppose \( H \) is regular. Then since \( L(0^*1^*) \) is regular, and regular languages are closed under intersection, \( H' \) also would be regular. But we know \( H' \) is not regular, a contradiction.
Non-regularity via closure properties

General recipe:

Apply closure properties

$L_1$  
$L_2$  
$L_n$  
$L_?$

KNOWN REGULAR

UNKNOWN

$L_{\text{non-regular}}$
Examples

$L = \{0^k1^k \mid k \geq 1\}$
Careful with closure!

\[ L' = \{0^k1^k \mid k \geq 0\} \]

Complement of \( L \) (\( \overline{L} \)) is also not regular.

But \( L \cup \overline{L} = (0 + 1)^* \) which is regular.

In general, always use closure in forward direction, (i.e \( L \) and \( L' \) are regular, therefore \( L \cup L' \) is regular. )

In particular, regular languages are not closed under subset/superset relations.
Proving non-regularity: Summary

• Method of distinguishing suffixes. To prove that $L$ is non-regular find an infinite fooling set.

• Closure properties. Use existing non-regular languages and regular languages to prove that some new language is non-regular.

• **Pumping lemma.** We did not cover it but it is sometimes an easier proof technique to apply, but not as general as the fooling set technique.