Today's learning goals

- Explain the limits of the class of regular languages
- Justify why the Pumping Lemma is true
- Apply the Pumping Lemma in proofs of nonregularity
- Identify some nonregular sets
All roads lead to … regular sets?

Are there any languages over \{0,1\} that are not regular?

A. Yes: a language that is recognized by an NFA but not any DFA.

B. Yes: there is some infinite language of strings over \{0,1\} that is not described by any regular expression.

C. No: all languages over \{0,1\} are regular because that's what it means to be a language.

D. No: for each set of strings over \{0,1\}, some DFA recognizes that set.

E. I don't know.
All languages over $\Sigma$

Regular languages over $\Sigma$

Finite languages over $\Sigma$
Counting

- **Fact:** a countable union of countable sets is countable.
- **Fact:** $\{0,1\}^*$ is countably infinite. $X^*$ is countably infinite when $X$ is finite.
- **Fact:** the set of subsets of a countably infinite set is uncountable.

- **Fact:** there are countably many DFA with $\Sigma=\{0,1\}$
- **Fact:** there are countably many regular languages over $\{0,1\}$
Counting

- Fact: A countable union of countable sets is countable.
- Fact: $\{0,1\}$ is countably infinite. $X^*$ is countably infinite when $X$ is finite.
- Fact: The set of subsets of a countably infinite set is uncountable.
- Fact: There are countably many DFA with $\Sigma = \{0,1\}$
- Fact: There are countably many regular languages over $\{0,1\}$

Uncountably many languages over $\{0,1\}$

Countably many regular languages over $\{0,1\}$
Birds' eye view

- All languages over $\Sigma$
- Regular languages over $\Sigma$
- Finite languages over $\Sigma$
Proving nonregularity

How can we prove that a set is non-regular?
A. Try to design a DFA that recognizes it and, if the first few attempts don't work, conclude there is none that does.
B. Prove that it's a strict subset of some regular set.
C. Prove that it's the union of two regular sets.
D. Prove that its complement is not regular.
E. I don't know.
Where we stand

• There exist non-regular sets.

• If we know that some sets are not regular, we can conclude others are \textbf{also} not regular \textit{judiciously reasoning using closure properties of class of regular languages}.

• No example of a specific regular set … \textit{yet}.
Bounds on DFA

• in DFA, memory = states

• Automata can only "remember"…
  • …finitely far in the past
  • …finitely much information

• If a computation path visits the same state more than once, the machine can't tell the difference between the first time and future times it visited that state.
Example!

\( \{ 0^n1^n \mid n \geq 0 \} \)

What are some strings in this set?  
What are some strings not in this set?

Compare to \( L(0^*1^*) \)  
Design a DFA? NFA?
Example!

\[ \{ 0^n1^n \mid n \geq 0 \} \]

What are some strings in this set?
What are some strings not in this set?

Compare to \( L(0^*1^*) \)
Design a DFA? NFA?
Pumping

- Focus on computation path through DFA
Pumping

- Focus on computation path through DFA
Pumping

- Focus on computation path through DFA

Idea: if one long string is accepted, then many other strings have to be accepted too.
Pumping Lemma

If $A$ is a regular language, then there is a number $p$ (the pumping length) where, if $s$ is any string in $A$ of length at least $p$, then $s$ may be divided into three pieces, $s = xyz$ such that

- $|y| > 0$, and
- for each $i \geq 0$, $xy^iz \in A$,
- $|xy| \leq p$. 
Pumping Lemma

If $A$ is a regular language, then there is a number $p$ (the pumping length) where, if $s$ is any string in $A$ of length at least $p$, then $s$ may be divided into three pieces, $s = x y z$ such that:

- $|y| > 0$, and
- for each $i \geq 0$, $xy^iz \in A$,
- $|xy| \leq p$.
Negation

- Pumping lemma "There is \( p \), where \( p \) is a pumping length for \( L \)"

- Given a specific number \( p \), it being a pumping length for \( L \) means

\[
\forall w \left( \left( |w| \geq p \land w \in L \right) \rightarrow \exists x \exists y \exists z \left( w = xyz \land |y| > 0 \land |xy| \leq p \land \forall i \left( xy^i z \in L \right) \right) \right)
\]

- So \( p \) not being a pumping length of \( L \) means

\[
\exists w \left( |w| \geq p \land w \in L \land \forall x \forall y \forall z \left( \left( w = xyz \land |y| > 0 \land |xy| \leq p \right) \rightarrow \exists i \left( xy^i z \notin L \right) \right) \right)
\]
Proof strategy

To prove that a language $L$ is not regular

- Assume towards a contradiction that it is.
- Use Pumping Lemma to give $p$, a pumping length for $L$
- Show that $p$ actually isn't a pumping length for $L$.
- \( \rightarrow \leftarrow \)
- Conclude that $L$ is not regular.
Using the Pumping Lemma

Claim: The set \( L = \{0^n1^n \mid n \geq 0\} \) is not regular.
Using the Pumping Lemma

Claim: The set $L = \{0^n1^n \mid n \geq 0\}$ is not regular.

Proof: Assume, towards a contradiction, that $L$ is regular.

Pumping Lemma gives property of all regular sets. Can we get a contradiction by assuming that the Pumping Lemma applies to this set?
Using the Pumping Lemma

Claim: The set \( L = \{0^n1^n \mid n \geq 0\} \) is not regular.

Proof:
Assume towards a contradiction \( L \) is regular.

So by Pumping Lemma, \( L \) has a pumping length, call it \( p \).

**FACT:** \( p \) is a pumping length for \( L \) (by definition).

**CLAIM:** \( p \) is not a pumping length for \( L \).

Conclude: contradiction!
Claim: The set $L = \{0^n1^n \mid n \geq 0\}$ is not regular.

Proof: ...In particular, this means that every string in $L$ that is of length $p$ or more can be "pumped".

Goal: pick a string $s$ in $L$ of length greater than or equal to $p$ such that any division of $s$ as $s = xyz$ with $|y| > 0$ and $|xy| \leq p$ gives some value $i \geq 0$ with $xy^iz$ not in $L$.

So we have a contradiction, and $L$ is not regular.
Claim: The set $L = \{0^n1^n \mid n \geq 0\}$ is not regular.

Proof: …

Goal: pick a string $s$ in $L$ of length greater than or equal to $p$ such that any division of $s$ as $s = xyz$ with $|y| > 0$ and $|xy| \leq p$ gives some value $i \geq 0$ with $xy^iz$ not in $L$.

Choose $s = 0^p1^p$. Consider any $s = xyz$ with $|y| > 0$, $|xy| \leq p$. 
Using the Pumping Lemma

Claim: The set \( L = \{0^n1^n \mid n \geq 0\} \) is not regular.

Proof: …

Goal: pick a string \( s \) in \( L \) of length greater than or equal to \( p \) such that any division of \( s \) as \( s = xyz \) with \(|y| > 0 \) and \(|xy| \leq p \) gives some value \( i \geq 0 \) with \( xy^iz \) not in \( L \).

Choose \( s = 0^p1^p \). Consider any \( s = xyz \) with \(|y| > 0 \), \(|xy| \leq p \).

Since \(|xy| \leq p \), \( x = 0^k \), \( y = 0^m \), \( z = 0^r1^p \) with \( k+m+r = p \), \( j > 0 \).
Using the Pumping Lemma

Claim: The set $L = \{0^n1^n \mid n \geq 0\}$ is not regular.

Proof: …

Goal: pick a string $s$ in $L$ of length greater than or equal to $p$ such that any division of $s$ as $s = xyz$ with $|y|>0$ and $|xy| \leq p$ gives some value $i \geq 0$ with $xy^iz$ not in $L$

Choose $s = 0^p1^p$. Consider any $s = xyz$ with $|y|>0$, $|xy| \leq p$.

Since $|xy| \leq p$, $x = 0^k$, $y = 0^m$, $z = 0^r1^p$ with $k+m+r = p$, $j>0$.

Picking $i=0$: $xy^iz = xz = 0^k0^m1^p = 0^{k+m}1^p$, not in $L$!
Using the Pumping Lemma

Claim: The set $L = \{0^n1^n \mid n \geq 0\}$ is not regular.

Proof: …

Goal: pick a string $s$ in $L$ of length greater than or equal to $p$ such that any division of $s$ as $s = xyz$ with $|y| > 0$ and $|xy| \leq p$ gives some value $i \geq 0$ with $xy^iz$ not in $L$

Choose $s = 0^p1^p$. Consider any $s = xyz$ with $|y| > 0$, $|xy| \leq p$.

Since $|xy| \leq p$, $x = 0^k$, $y = 0^m$, $z = 0^r1^p$ with $m+n+r = p$, $j > 0$.

Picking $i = 0$: $xy^iz = xz = 0^k0^m1^p = 0^{k+m}1^p$, not in $L$! This is a contradiction with the Pumping Lemma applied to $L$, so $L$ must not be regular.
Another example

Claim: The set \( \{a^m b^n a^m \mid m, n \geq 0\} \) is not regular.

Proof: …Consider the string \( s = \ldots \)

You must pick \( s \) carefully: we want \(|s| \geq p\) and \( s \) in \( L \). Now we will prove a contradiction with the statement "s can be pumped"

Which choices of \( s \) cannot be used to complete the proof?

A. \( s = a^p b^p \)  
B. \( s = ab^p a \)  
C. \( s = a^p b^p a^p \)  
D. \( s = a^p ba^p \)

E. None of the above (all of these choices work).
Another example

Claim: The set \( \{a^m b^m a^n \mid m, n \geq 0\} \) is not regular.

Proof: \( \ldots \text{Consider the string } s = \ldots \) 

You must pick \( s \) carefully: we want \( |s| \geq p \) and \( s \) in \( L \). Now we will prove a contradiction with the statement "\( s \) can be pumped".

Consider an arbitrary choice of \( x, y, z \) such that \( s = xyz, \ |y| > 0, \ |xy| \leq p \). \textbf{This means that...} What properties are guaranteed about \( x, y, z \)?

\( \text{Consider } i = \ldots \) In this case, \( xy^i z = \ldots \), which is not in \( L \), a contradiction with the Pumping Lemma applying to \( L \) and so \( L \) is not regular.
And another

Claim: The set \( \{ w \, w^R \mid w \text{ is a string over } \{0,1\} \} \) is not regular.

Proof: …Consider the string \( s = \ldots \) You must pick \( s \) carefully: we want \( |s| \geq p \) and \( s \) in \( L \). Now we will prove a contradiction with the statement "\( s \) can be pumped" Consider \( i = \ldots \)

Which \( s \) and \( i \) let us complete the proof?

A. \( s = 0^p0^p \), \( i=2 \)  
B. \( s = 0110 \), \( i=0 \)  
C. \( s = 0^p110^p \), \( i=1 \)  
D. \( s = 1^p001^p \), \( i=3 \)  
E. I don't know
How do we choose $i$?

Claim: The set $\{0^j1^k \mid j, k \geq 0 \text{ and } j \geq k \}$ is not regular.

Proof: …Consider the string $s = \ldots$

You must pick $s$ carefully: we want $|s| \geq p$ and $s$ in $L$. Now we will prove a contradiction with the statement "$s$ can be pumped" Consider $i = \ldots$

Which $s$ and $i$ let us complete the proof?

A. $s = 0^p1^p$, $i=2$  B. $s = 0^p1^p$, $i=p$  C. $s = 0^p1^p$, $i=1$  D. $s = 0^p1^p$, $i=0$

E. I don't know
Regular sets: not the end of the story

- Many **nice / simple / important** sets are not regular
- Limitation of the finite-state automaton model
  - Can't "count"
  - Can only remember finitely far into the past
  - Can't backtrack
  - Must make decisions in "real-time"
- We know computers are more powerful than this model…

*Which conditions should we relax?*
The next model of computation

• **Idea:** allow *some* memory of unbounded size

• **How?**
  - Generalization of regular expressions → **Context-free grammars**
  - Generalization for DFA → **Pushdown Automata**