

Week10 friday

Model of Computation	Class of Languages
<p>Deterministic finite automata: formal definition, how to design for a given language, how to describe language of a machine? Nondeterministic finite automata: formal definition, how to design for a given language, how to describe language of a machine? Regular expressions: formal definition, how to design for a given language, how to describe language of expression? <i>Also:</i> converting between different models.</p>	<p>Class of regular languages: what are the closure properties of this class? which languages are not in the class? using pumping lemma to prove nonregularity.</p>
<p>Push-down automata: formal definition, how to design for a given language, how to describe language of a machine? Context-free grammars: formal definition, how to design for a given language, how to describe language of a grammar?</p>	<p>Class of context-free languages: what are the closure properties of this class? which languages are not in the class?</p>
<p>Turing machines that always halt in polynomial time</p> <p>Nondeterministic Turing machines that always halt in polynomial time</p>	<p>P</p> <p>NP</p>
<p>Deciders (Turing machines that always halt): formal definition, how to design for a given language, how to describe language of a machine?</p>	<p>Class of decidable languages: what are the closure properties of this class? which languages are not in the class? using diagonalization and mapping reduction to show undecidability</p>
<p>Turing machines formal definition, how to design for a given language, how to describe language of a machine?</p>	<p>Class of recognizable languages: what are the closure properties of this class? which languages are not in the class? using closure and mapping reduction to show unrecognizability</p>

Given a language, prove it is regular

Strategy 1: construct DFA recognizing the language and prove it works.

Strategy 2: construct NFA recognizing the language and prove it works.

Strategy 3: construct regular expression recognizing the language and prove it works.

“Prove it works” means ...

Example: $L = \{w \in \{0, 1\}^* \mid w \text{ has odd number of 1s or starts with } 0\}$

Using NFA

Using regular expressions

Example: Select all and only the options that result in a true statement: “To show a language A is not regular, we can...”

- a. Show A is finite
- b. Show there is a CFG generating A
- c. Show A has no pumping length
- d. Show A is undecidable

Example: What is the language generated by the CFG with rules

$$\begin{aligned} S &\rightarrow aSb \mid bY \mid Ya \\ Y &\rightarrow bY \mid Ya \mid \varepsilon \end{aligned}$$

Example: Prove that the language $T = \{\langle M \rangle \mid M \text{ is a Turing machine and } L(M) \text{ is infinite}\}$ is undecidable.

Example: Prove that the class of decidable languages is closed under concatenation.