Theory of Computation

Pushdown Automata and CFGs

Barath Raghavan
CS 361 Fall 2009
Williams College
MISC

Homework 1 back
Quiz 1 due tomorrow
Is the following language regular?

\[ L = \{ 0^n 1^n \mid n \geq 0 \} \]

\[ \Sigma = \{ 0, 1 \} \]
What makes a language Regular

NON
DFAs/NFAs cannot store information.
Suppose you have a stack on which to store information.
How might you recognize \( L \)?

\[
L = \{ 0^n 1^n \mid n \geq 0 \} \\
\Sigma = \{ 0, 1 \}
\]
NFAs $Q$ set of states
NFAs $Q$ set of states
$\Sigma$ the alphabet
NFAs

$Q$ set of states

$\Sigma$ the alphabet

$\delta : Q \times \Sigma_e \rightarrow P(Q)$ transition function
NFAs

- $Q$: set of states
- $\Sigma$: the alphabet
- $\delta: Q \times \Sigma_e \rightarrow P(Q)$: transition function
- $q_0 \in Q$: start state
NFAs

- $Q$: set of states
- $\Sigma$: alphabet
- $\delta: Q \times \Sigma \rightarrow P(Q)$: transition function
- $q_0 \in Q$: start state
- $F \subseteq Q$: set of final states
Pushdown automata: push and pop a stack.
NPDAs

\[ Q \times \Sigma, \Gamma \]

\[ Q \times \Sigma_\varepsilon \times \Gamma_\varepsilon \rightarrow P(Q \times \Gamma_\varepsilon) \]

\[ q_0 \in Q \]

\[ F \subseteq Q \]

set of states

input, stack alphabet

transition function

start state

set of final states
On this input and this on the stack

Push this on the stack

\[ 0, \varepsilon \rightarrow 0 \]

\[ 1, 0 \rightarrow \varepsilon \]

\[ \varepsilon, \varepsilon \rightarrow \$ \]
\[ L = \{0^n 1^n \mid n \geq 0\} \]
\[ \Sigma = \{0, 1\} \quad \Gamma = \{0, \$\} \]
What NPDA recognizes:

\[ L = \{0^i 1^j 2^k \mid i, j, k \geq 0 \text{ and } i = j \text{ or } i = k\} \]

\[ \Sigma = \{0, 1, 2\} \]
$L = \{0^i1^j2^k \mid i, j, k \geq 0 \text{ and } i = j \text{ or } i = k\}$
Build an NPDA that recognizes:

\[ L = \{ ww^R \} \]

\[ \Sigma = \{ 0, 1 \} \]
\[ L = \left\{ \omega \omega^R \right\} \]
Build an NPDA that recognizes:

\[ L = \{ w \mid w \text{ has the same number of 0s and 1s} \} \]

\[ \Sigma = \{0, 1\} \]
$L = \{ w \mid w \text{ has the same number of 0s and 1s} \}$
How can we define an NPDA’s language?
What CFG generates this language?

\[
L = \{0^n 1^n \mid n \geq 0\}
\]

\[
\Sigma = \{0, 1\}
\]
\[ L = \{0^n1^n \mid n \geq 0\} \]
\[ \Sigma = \{0, 1\} \]

**Variables**

- \(A\)
- \(B\)

**Rules**

- \(A \rightarrow 0A1\)
- \(A \rightarrow B\)
- \(B \rightarrow \varepsilon\)

**Terminals**
Grammar $G$:

$$
A \rightarrow 0A1 \\
A \rightarrow B \\
B \rightarrow \varepsilon
$$

$L(G) = \{0^n1^n \mid n \geq 0\}$
Grammar $G$:

$A \rightarrow 0A1$

$A \rightarrow B$

$B \rightarrow \varepsilon$

Derivation of a string:

$A \Rightarrow 0A1 \Rightarrow 00A11 \Rightarrow 000B111 \Rightarrow 000111$
What CFG generates this language?

$L = \{ww^R\}$

$\Sigma = \{0, 1\}$
\[ L = \{ w w^R \} \]

Grammar \( G \):

\[
\begin{align*}
A & \rightarrow 0A0 \mid 1A1 \\
A & \rightarrow \varepsilon
\end{align*}
\]
What CFG generates this language?

\[ L = \{ w \mid w \text{ has the same number of 0s and 1s} \} \]

\[ \Sigma = \{0, 1\} \]
\[ L = \{ w \mid w \text{ has the same number of 0s and 1s} \} \]

**Grammar G:**

\[
\begin{align*}
A & \rightarrow 0A1A \mid 1A0A \\
A & \rightarrow \varepsilon
\end{align*}
\]
Reading: Sipser 2.1, 2.2