MTH-682 Automata
Assignment (2): Context-Free Languages

Mostafa Hassanein

20 November 2025

2.1

Recall the CFG G4 that we gave in Example 2.4. For convenience, let's rename its variables with single letters as follows.

$$
\begin{aligned}
E &\to E + T | T \\
T &\to T x F | F \\
F &\to (E) | a
\end{aligned}
$$

Give parse trees and derivations for each string.

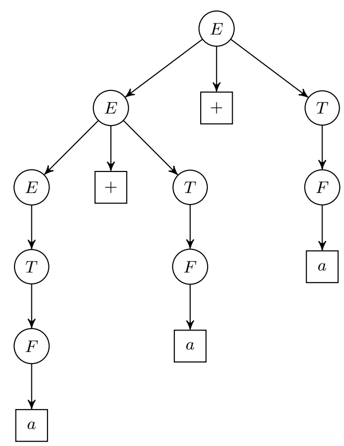
c. $a + a + a$

Solution:

Derivation:
We derive the string $a + a + a$ from the start symbol E as follows:

$$
\begin{aligned}
E &\Rightarrow E + T \\
&\Rightarrow E + T + T \\
&\Rightarrow T + T + T \\
&\Rightarrow F + T + T \\
&\Rightarrow a + T + T \\
&\Rightarrow a + F + T \\
&\Rightarrow a + a + T \\
&\Rightarrow a + a + F \\
&\Rightarrow a + a + a
\end{aligned}
$$

Parse Tree:



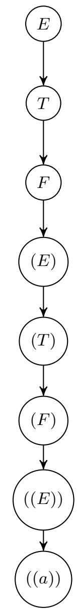
c. $((a))$

Solution:

Derivation:
We derive the string $((a))$ from the start symbol E as follows:

$$
\begin{aligned}
E &\Rightarrow T \\
&\Rightarrow F \\
&\Rightarrow (E) \\
&\Rightarrow (T) \\
&\Rightarrow (F) \\
&\Rightarrow ((E)) \\
&\Rightarrow ((T)) \\
&\Rightarrow ((F)) \\
&\Rightarrow ((a))
\end{aligned}
$$

Parse Tree:



2.4

Give context-free grammars that generate the following languages. In all parts, the alphabet Σ is $\{0, 1\}$.

c. $\{w|$ the length of w is odd$\}$

Solution:

$$S \to ASA|0|1$$
$$A \to 0|1$$

c. The empty set

Solution:

$$S \to S$$

This language has a single rule that infinitely recurses and never terminates to any terminal symbols. Therefore, no words belong to this language.

2.5

Give informal descriptions and state diagrams of pushdown automata for the languages in Exercise 2.4.

c. $\{w|$ the length of w is odd$\}$

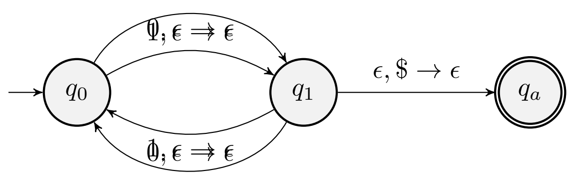
<u>Solution:</u>

Informal Description:

The pushdown automaton reads input symbols (0 or 1) one at a time and uses two states to track the parity of the number of symbols read. It starts in state q_0 (representing an even count, since 0 symbols have been read). Each input symbol causes a transition that toggles between states q_0 and q_1. After reading all input, if the automaton is in state q_1 (odd count), it transitions to the accept state q_{accept}.

The stack is not essential for this language, but we include it for the formal PDA definition. We use a bottom-of-stack marker \$ to detect when input is complete.

State Diagram:



Formal Description:

- States: $Q = \{q_0, q_1, q_a\}$
- Start state: q_0
- Accept state: q_a
- Stack alphabet: $\Gamma = \{\$\}$
- Initial stack symbol: \$
- Transitions:

$$\begin{aligned}
\delta(q_0, 0, \epsilon) &= \{(q_1, \epsilon)\} \\
\delta(q_0, 1, \epsilon) &= \{(q_1, \epsilon)\} \\
\delta(q_1, 0, \epsilon) &= \{(q_0, \epsilon)\} \\
\delta(q_1, 1, \epsilon) &= \{(q_0, \epsilon)\} \\
\delta(q_1, \epsilon, \$) &= \{(q_a, \epsilon)\}
\end{aligned}$$

The automaton alternates between states q_0 and q_1 for each symbol read. It accepts if and only if it ends in state q_1 (odd number of symbols) and can transition to q_a by popping the bottom marker \$.

f.

<u>Solution:</u>

2.11

Convert the CFG G4 given in Exercise 2.1 to an equivalent PDA, using the procedure given in Theorem 2.20.

Solution:

2.13

Let $G = (V, \Sigma, R, S)$ be the following grammar. $V = S, T, U$; $\Sigma = 0, \#$; and R is the set of rules:

$$S \rightarrow TT|U$$
$$T \rightarrow 0T|T0|\#$$
$$U \rightarrow 0U00|\#$$

a. Describe $L(G)$ in English.
b. Prove that $L(G)$ is not regular.

a.

<u>Solution:</u>

b.

<u>Solution:</u>

2.14

Convert the following CFG into an equivalent CFG in Chomsky normal form, using the procedure given in Theorem 2.9.

$$A \rightarrow BAB|B|\epsilon$$
$$B \rightarrow 00|\epsilon$$

Solution:

2.23

Let $D = \{xy|\, x, y \in \{0,1\}^*$ and $|x| = |y|$ but $x \neq y\}$.
Show that D is a context-free language.

Solution:

2.24

Let $E = \{a^i b^j \mid i \neq j \text{ and } 2i \neq j\}$.
Show that E is a context-free language.

Solution:

2.26

Show that if G is a CFG in Chomsky normal form, then for any string $w \in L(G)$ of length n $n \geq 1$, exactly $2n - 1$ steps are required for any derivation of w.

Solution:

2.27

Let $G = (V, \Sigma, R, < STMT >)$ be the following grammar.

$$
\begin{aligned}
< STMT > &\to < ASSIGN > \mid < IF-THEN > \mid < IF-THEN-ELSE > \\
< IF-THEN > &\to if condition then < STMT > \\
< IF-THEN-ELSE > &\to if condition then < STMT > else < STMT > \\
< ASSIGN > &\to a := 1
\end{aligned}
$$

$\Sigma = \{if,\ condition,\ then,\ else,\ a := 1\}$
$V = \{< STMT >, < IF-THEN >, < IF-ELSE-THEN >, < ASSIGN >\}$

G is a natural-looking grammar for a fragment of a programming language, but G is ambiguous.

a. Show that G is ambiguous.
b. Give a new unambiguous grammar for the same language.

a.

Solution:

b.

Solution:

30

Use the pumping lemma to show that the following languages are not context free.

d.

$$L = \left\{ t1\#t2\#\ldots\#t_k \middle|\, k \geq 2, \forall i\; t_i \in \{a,b\}^* \;\wedge\; \exists i,j : (t_i = t_j \;\wedge\; i \neq j) \right\}$$

<u>Solution:</u>

2.31

Let B be the language of all palindromes over $\{0, 1\}$ containing equal numbers of 0s and 1s. Show that B is not context free.

Solution: