1. Recall the CFG G_4 that we gave in Example 2.4. For convenience, let's rename its variables with single letters as follows.

$$E \to E + T \mid T$$
$$T \to T \times F \mid F$$
$$F \to (E) \mid a$$

Give parse trees and derivations for each string.
a. a
b. a+a
c. a+a+a
d. ((a))
a. $E \Rightarrow T \Rightarrow F \Rightarrow a$

b. $E \Rightarrow E + T \Rightarrow T + T \Rightarrow F + T \Rightarrow F + F \Rightarrow a + F \Rightarrow a + a$

c. $E \Rightarrow E + T \Rightarrow E + T + T \Rightarrow T + T + T \Rightarrow F + T + T \Rightarrow F + F + T \Rightarrow F + F + F \Rightarrow a + F + F \Rightarrow a + a + F \Rightarrow a + a + a$

d. $E \Rightarrow T \Rightarrow F \Rightarrow (E) \Rightarrow (T) \Rightarrow (F) \Rightarrow ((E)) \Rightarrow ((T)) \Rightarrow ((F)) \Rightarrow ((a))$

2. Give context-free grammars that generate the following languages. In all parts, the alphabet Σ is {0,1}.
b. $\{w \mid w$ starts and ends with the same symbol$\}$
c. $\{w \mid$ the length of w is odd$\}$
e. $\{w \mid w = w^{\mathcal{R}}$, that is, w is a palindrome$\}$
f. The empty set

b. $S \to 0R0 \mid 1R1 \mid \varepsilon$
$R \to 0R \mid 1R \mid \varepsilon$
c. $S \to 0 \mid 1 \mid 00S \mid 01S \mid 10S \mid 11S$
e. $S \to 0S0 \mid 1S1 \mid 0 \mid 1 \mid \varepsilon$
f. $S \to S$

3. Give informal descriptions and state diagrams of pushdown automata for the languages in Exercise 2.
(**b.**, **c.**, **e.**, **f.**)

(a) This set is regular. The PDA scans the string and keep track of the first and last symbol in its finite control. If they are the same, it accepts, otherwise it rejects.

(b) This set is regular. The PDA scans the string and keep track of the length (modulo 2) using its finite control. If the length is 1 (modulo 2) it accepts, otherwise it rejects.

(c) The PDA begins by scanning across the string and pushing each symbol onto its stack. At some point it nondeterministically guesses when it has reached the middle. It also nondeterministically guesses if the string has odd length or even length. If it guessed even, then it pushes the current symbol it's reading onto the stack (recall that the PDA has guessed that this symbol is the middle symbol). If it guesses that string has odd length, it goes to the next input symbol without changing the stack. Now, it scans the rest of the string, and it compares each symbol it scans to the symbol on the top of the stack. If they are the same, it pops the stack, and continues scanning. If they are different, it rejects. If the stack is empty before it finishes reading all the input, it accepts. If the stack becomes empty just after it reaches the end of the input then it accepts. In all other cases, it rejects.

(d) The PDA simply rejects immediately.

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

$$A \to BAB \mid B \mid \varepsilon$$
$$B \to 00 \mid \varepsilon$$

The equivalent CFG in Chomsky normal form is given as follows.

$$S_0 \to AB \mid CC \mid BA \mid BD \mid BB \mid \varepsilon$$
$$A \to AB \mid CC \mid BA \mid BD \mid BB$$
$$B \to CC$$
$$C \to 0$$
$$D \to AB$$

5. Convert the CFG G_4 given in Exercise 1 to an equivalent PDA, using the procedure given in Theorem 2.20.

Informal description of a PDA that recognizes the CFG in Exercise 1:

(a) Place the marker symbol \$ and the start variable E on the stack.
(b) Repeat the following steps forever.
(c) If the top of stack is the variable E, pop it and nondeterministically push either $E+T$ or T into the stack.
(d) If the top of stack is the variable T, pop it and nondeterministically push either TxF or F into the stack.
(e) If the top of stack is the variable F, pop it and nondeterministically push either (E) or a into the stack.
(f) If the top of stack is a terminal symbol, read the next symbol from the input and compare it to the terminal in the stack. If they match, repeat. If they do not match, reject on this branch of the nondeterminism.
(g) If the top of stack is the symbol \$, enter the accept state. Doing so accepts the input if it has all been read.

The formal definition of the equivalent PDA is $(Q, \Sigma, \Gamma, \delta, q_1, F)$, where $Q = \{q_1, q_2\}$; $\Sigma = \{+, x, (,), a\}$; and $\Gamma = \{E, T, F\} \cup \Sigma$; $F = \{q_2\}$. The transition function $\delta : Q \times \Sigma_\varepsilon \times \Gamma_\varepsilon \to \mathcal{P}(Q \times \Gamma_\varepsilon)$ is given as follows.

$$\delta(q,x,y) = \begin{cases} \{(q_2, \epsilon)\} \\ \text{if } q = q_1, x = \varepsilon, y = \$ \\ \\ \{(q_1, E+T), (q_1, T)\} \\ \text{if } q = q_1, x = \varepsilon, y = E \\ \\ \{(q_1, T \times F), (q_1, F)\} \\ \text{if } q = q_1, x = \varepsilon, y = T \\ \\ \{(q_1, (E)), (q_1, a)\} \\ \text{if } q = q_1, x = \varepsilon, y = F \\ \\ \{(q_1, \varepsilon)\} \\ \text{if } q = q_1, x = y \end{cases}$$

6. This exercise concerns TM M_2, whose description and state diagram appear in Example 3.7. In each of the parts, give the sequence of configurations that M_2 enters when started on the indicated input string.

   a. 0
   b. 00
   c. 000
   d. 000000

   (a) $q_1 0, \sqcup q_2 \sqcup, \sqcup\sqcup q_{\text{accept}}$
   (b) $q_1 00, \sqcup q_2 0, \sqcup x q_3 \sqcup, \sqcup q_5 x \sqcup, q_5 \sqcup x \sqcup, \sqcup q_2 x \sqcup, \sqcup x q_2 \sqcup, \sqcup x \sqcup q_{\text{accept}}$
   (c) $q_1 000, \sqcup q_2 00, \sqcup x q_3 0, \sqcup x 0 q_4 \sqcup, \sqcup x 0 \sqcup q_{\text{reject}}$
   (d) $q_1 000000, \sqcup q_2 00000, \sqcup x q_3 00000, \sqcup x 0 q_4 000, \sqcup x 0 x q_3 00, \sqcup x 0 x 0 q_4 0, \sqcup x 0 x 0 x q_3 \sqcup, \sqcup x 0 x 0 q_5 x \sqcup, \sqcup x 0 x q_5 0 x \sqcup, \sqcup x 0 q_5 x 0 x \sqcup, \sqcup x q_5 0 x 0 x \sqcup, q_5 \sqcup x 0 x 0 x \sqcup, \sqcup q_2 x 0 x 0 x \sqcup, \sqcup x q_2 0 x 0 x \sqcup, \sqcup x x q_3 x 0 x \sqcup, \sqcup x x x 0 q_4 x \sqcup, \sqcup x x x 0 x q_4 \sqcup, \sqcup x x x 0 x \sqcup q_{\text{reject}}$

7. This exercise concerns TM M_1, whose description and state diagram appear in Example 3.9. In each of the parts, give the sequence of configurations that M_1 enters when started on the indicated input string.

   a. 11
   b. 1#1
   d. 10#11
   e. 10#10

   (a) $q_1 11, \sqcup q_3 1, \sqcup 1 q_3 \sqcup, \sqcup 1 \sqcup q_{\text{reject}}$
   (b) $q_1 1\#1, \sqcup q_3 \#1, \sqcup \# q_5 1, \sqcup \# 1 q_5 \sqcup, \sqcup \# q_7 1 \sqcup, \sqcup q_7 \# 1 \sqcup, q_7 \sqcup 0 \# 1 \sqcup, \sqcup q_9 \# 1 \sqcup, \sqcup \# q_{11} 1 \sqcup, \sqcup q_{12} \# x \sqcup, q_{12} \sqcup 0 \# x \sqcup, \sqcup q_{13} \# x \sqcup, \sqcup \# q_{14} x \sqcup, \sqcup \# x q_{14} \sqcup, \sqcup \# x \sqcup q_{\text{accept}}$
   (c) $q_1 1\#\#1, \sqcup q_3 \#\#1, \sqcup \# q_5 \#1, \sqcup \#\# q_{\text{reject}} 1$
   (d) $q_1 10\#11, \sqcup q_3 0\#11, \sqcup 0 q_3 \#11, \sqcup 0 \# q_5 11, \sqcup 0 \# 1 q_5 1, \sqcup 0 \# 11 q_5 \sqcup, \sqcup 0 \# 1 q_7 1 \sqcup, \sqcup 0 \# q_7 11 \sqcup, \sqcup 0 q_7 \# 11 \sqcup, \sqcup q_7 \sqcup 0 \# 11 \sqcup, \sqcup q_9 0 \# 11 \sqcup, \sqcup 0 q_9 \# 11 \sqcup, \sqcup 0 \# q_{11} 11 \sqcup, \sqcup 0 q_{12} \# x 1 \sqcup, \sqcup q_{12} 0 \# x 1 \sqcup, q_{12} \sqcup 0 \# x 1 \sqcup, \sqcup q_{13} 0 \# x 1 \sqcup, \sqcup x q_8 \# x 1 \sqcup, \sqcup x \# q_{10} x 1 \sqcup, \sqcup x \# x q_{10} 1 \sqcup, \sqcup x \# x 1 q_{\text{reject}}$
   (e) $q_1 10\#10, \sqcup q_3 0\#10, \sqcup 0 q_3 \#10, \sqcup 0 \# q_5 10, \sqcup 0 \# 1 q_5 0, \sqcup 0 \# 10 q_5 \sqcup, \sqcup 0 \# 1 q_7 0 \sqcup, \sqcup 0 \# q_7 10 \sqcup, \sqcup 0 q_7 \# 10 \sqcup, \sqcup q_7 \sqcup 0 \# 10 \sqcup, \sqcup q_9 0 \# 10 \sqcup,$

$\sqcup\ 0q_9\#10\sqcup, \sqcup 0\#q_{11}10\sqcup, \sqcup 0q_{12}\#x0\sqcup,$
$\sqcup\ q_{12}0\#x0\sqcup, q_{12}\sqcup 0\#x0\sqcup, \sqcup q_{13}0\#x0\sqcup,$
$\sqcup\ xq_8\#x0\sqcup, \sqcup x\#q_{10}x0\sqcup, \sqcup x\#xq_{10}0\sqcup,$
$\sqcup x\#x0\sqcup, \sqcup x\#q_{12}xx\sqcup, \sqcup xq_{12}xx\sqcup, \sqcup q_{12}xx\sqcup,$
$\sqcup\ q_{13}xx\sqcup, \sqcup xq_{13}xx\sqcup, \sqcup x\#q_{14}xx\sqcup,$
$\sqcup\ x\#xq_{14}x\sqcup, \sqcup x\#xxq_{14}\sqcup, \sqcup x\#xx\sqcup q_{\text{accept}}$