1 Exercise 5.5

(a) The PDA accepts odd-length strings whose middle symbol is $a$ and whose other letters are $a$s and $b$s. Its diagram is below.

(b)
2 Exercise 5.10

Let $L = L(M)$ be a regular language defined through an FA $M$. Since our PDA (call it $M'$) may only have two states $q'_0, q'_1$, exactly one of them should be an accepting state, and one of them should be the starting state. Suppose $q'_0$ is the starting state. If $\Lambda \in L$, then $q'_0$ has to be also the accepting state, otherwise, $q'_1$ will be the accepting state. Since $M'$ has to simulate $M$, which may contain more than 2 states, we have to store that information somewhere. So, we will make the stack alphabet out of the set $Q$ of states in $M$. For every transition

$$q_1 \xrightarrow{a} q_2$$

in $M$, we create a transition

$$q'_i \xrightarrow{a,q_1/q_1} q'_j$$

in machine $M'$, where we select $i, j$ as 0 or 1 to match the acceptance of states $q_1$ and $q_2$ in $M$. Machine $M'$ simulates automaton $M$ by keeping its current state at the top of the stack.
3 CYK Exercise

Since \( S \) appears in the last cell of the table, we know that \( babaa \) is generated by the grammar. To be able to retrieve a derivation tree from the table, we would need to keep more information in each cell. Specifically, for each non-terminal in cell, we write which production is used for it, and what partition of the subword is obtained. For example, we show below the augmented version of the above table:

To retrieve the derivation tree from the above table, we have to retrace the productions used, starting from the initial \( S \) in the bottom cell. The productions used in the correct derivation have been underlined in the table.
4 Pumping Lemma Exercise

(a) $L = \{a^r b^s c^t : r > s > t\}$
For the sake of contradiction, suppose that $L$ is context-free. Let $n$ be the constant from the Pumping Lemma. Take $u = a^{n+1}b^n c^{n-1}$. Since $|u| \geq n$, by the Pumping Lemma we know that $u = vwxyz$, as in the lemma. Moreover, since $|wxy| \leq n$, we have two cases:

1. $wy = a^k b^l$, where $k, l \geq 0$. Then, $vzx = a^{n+1-k} b^{n-l} c^{n-1}$ has to be in $L$, according to the Pumping Lemma. But notice that either $l > 0$, in which case $n - l \leq n - 1$, or $k > 0 = l$, which means that $n + 1 - k \leq n = n - l$, so the conditions of language $L$ will be violated, obtaining a contradiction.

2. $wy = b^k c^l$, where $l > k \geq 0$. Then $u_2 = vwxz^2$ has to be in $L$, according to the Pumping Lemma. But notice that either $k > 0$, in which case $\#(a(u_2)) = n + k \geq \#(a(u_2))$, or $k = 0$, which means that $\#(a(u_2)) = n - 1 + l \geq \#(b(u_2))$, so the conditions of language $L$ will be violated, getting contradiction.

(b) $L = \{ab^n ab^n ab^n : n \geq 0\}$
For the sake of contradiction, suppose that $L$ is context-free. Let $n$ be the constant from the Pumping Lemma. Take $u = ab^n ab^n ab^n$. Since $|u| \geq n$, by the Pumping Lemma we know that $u = vwxz$, as in the lemma. Moreover, since $|wxy| \leq n$, we have two cases:

1. $wy$ contains at least one $a$. Word $vzx$ has to be in $L$, according to the Pumping Lemma. But notice that since $u$ had exactly 3 letters $a$ and we removed at least one of them, then $vzx$ will have less than 3 $a$’s, so the conditions of language $L$ will be violated, obtaining a contradiction.

2. $wxy = b^k$, where $n \geq k > 0$. Observe, that $wxy$ has to then be contained within one of the three blocks $b^n$. Without loss of generality, we can assume that it is the first block. According to Pumping Lemma, the word $vwxz^2 = ab^n ab^n ab^n$ has to be in the language, but the first block of $b$’s is larger than the other two, so the conditions of language $L$ will be violated, getting contradiction.
Exercise 6.5

(e) [Extra Credit] We will show that $L = \{x \in \{a, b\}^* \mid n_a(x) < n_b(x) < 2n_a(x)\}$ is a CFL by demonstrating a grammar for this language. First, let us describe three related languages:

1. $L_1 = \{x \in \{a, b\}^* \mid n_a(x) = n_b(x)\}$. We know from the previous homework, that $L_1$ has for instance this grammar:

   $$\begin{align*}
   S_1 & \rightarrow aS_1bS_1 \mid bS_1aS_1 \mid \Lambda.
   \end{align*}$$

   Each production in this grammar adds one $a$ and one $b$ to the word. We will call those the type 1 productions.

2. $L_2 = \{x \in \{a, b\}^* \mid n_b(x) = 2n_a(x)\}$. We can similarly obtain a grammar for this language:

   $$\begin{align*}
   S_2 & \rightarrow aS_2bS_2bS_2 \mid bS_2aS_2bS_2 \mid bS_2bS_2aS_2 \mid \Lambda.
   \end{align*}$$

   Here, we have productions that add two $b$'s for each $a$ - type 2 productions.

3. $L_{12} = \{x \in \{a, b\}^* \mid n_a(x) \leq n_b(x) \leq 2n_a(x)\}$. For this language, we get a grammar by combining the productions of both types:

   $$\begin{align*}
   S_{12} & \rightarrow B_1 \mid B_2 \mid \Lambda \\
   B_1 & \rightarrow aS_{12}bS_{12} \mid bS_{12}aS_{12} \\
   B_2 & \rightarrow aS_{12}bS_{12}bS_{12} \mid bS_{12}aS_{12}bS_{12} \mid bS_{12}bS_{12}aS_{12}.
   \end{align*}$$

Language $L_{12}$ is already very close to $L$, except the inequalities are not strict. This becomes a problem when a derivation consists of only productions using $B_1$, and never $B_2$, or vice versa. We can solve that by forcing the derivation to have first a type 1 production, then a type 2 production, and then we can essentially revert to the last grammar:

$$\begin{align*}
S & \rightarrow aS_2bS_2 \mid aS_2bS_2 \mid bS_2aS_2 \mid bS_2aS_2 \\
S_2 & \rightarrow B_2 \\
S_{12} & \rightarrow B_1 \mid B_2 \mid \Lambda.
\end{align*}$$
with non-terminals $B_1, B_2$ having the same productions as earlier described. This gives us language $L$.

(g) Language $L$ is a CFL because we know that it is the complement of the language of balanced parentheses, which is a DCFL, and DCFLs are closed under complement.

6 Exercise 6.9

(a) Below is a PDA for the language $L = \{a^i b^j c^k \mid i \geq j \text{ or } i \geq k\}$, showing that it is a CFL.

Let $R$ be the regular language corresponding to $a^* b^* c^*$. Moreover, let $L_2 = \{a^i b^j c^k \mid i < j \text{ and } i < k\}$. Then, the complement language satisfies the following formula:

$$L_2 = L' \cap R.$$  

For the sake of contradiction, suppose that $L'$ is a CFL. Then, $L_2$ has to be a CFL too, as an intersection of a CFL and a regular language. Let $n$ be the constant from the Pumping Lemma for language $L_2$. Take $u = a^{n-1} b^n c^n$. Since $|u| \geq n$, by the Pumping Lemma we know that $u = vwxyz$, as in the lemma. Moreover, since $|wxy| \leq n$, we have two cases:

1. $wy = a^k b^l$, where $k, l \geq 0$. Then, $vxz = a^{n-1-k} b^n \cdot c^n$ and $v w^2 x y^2 z = a^{n-1+k} b^{n+l} c^n$ both have to be in $L$, according to the Pumping Lemma. But notice that either $k > 0$, in which case $n-1+k \geq n$ (no less $a$'s than $c$'s in $v w^2 x y^2 z$), or $l > 0 = k$, which means that $n-l \leq n-1 = n-1-l$
(no less $a$’s than $b$’s in $vxz$), so the conditions of language $L$ will be 
violated, obtaining a contradiction.

2. $wy = b^k c^l$, where $l > k \geq 0$. Then $vzx = a^{n-1}b^{n-k}c^{n-l}$ has to be in $L$, 
according to the Pumping Lemma. But notice that since $l \geq 1$, there
is at least as many $a$’s as $c$’s in $vzx$ so the conditions of language $L$ will 
be violated, getting contradiction.

This proves that $L'$ is not a CFL.