6 Regular Grammars, PDA's, and Properties of CFL's

6.1 Find CFG's for the following regular expressions:

a) $00(1+0)^*1$

b) $101(101)^*010(010)^*$

c) $(11+010)^*11(00+11)^*$

6.2 For each of the following CFG's, give an NFA_ϵ accepting the language in question. (S is the start symbol in all cases.)

a) $S \to 0S \mid 1A \mid \epsilon$
$A \to 1A \mid \epsilon$

b) $S \to 01S \mid 00$

c) $S \to 10A \mid 00A$
$A \to 10A \mid 01B \mid 11$
$B \to 01B \mid 11$

6.3 A context free grammar $G = (N, \Sigma, P, S)$ is called *right-linear* if all its productions are of the form

$$A \to wB \quad \text{or} \quad A \to w,$$

where $w \in \Sigma^*$ and $A, B \in N$. A *left-linear* grammar is defined analogically. A grammar is *regular* if it is right- or left-linear.

Find a regular CFG which generates the language that is accepted by

a) the NFA in figure 13,

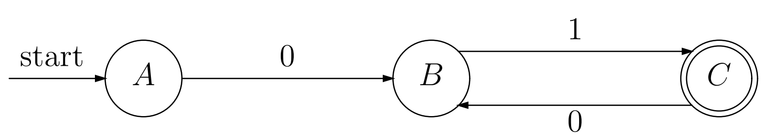

Figure 13: M_{13}

b) the NFA in figure 14.

6.4 Consider the following PDA M, where

States: $\{q_0, q_1, q_2\}$

Input alphabet: $\{a, b\}$

Stack alphabet: $\{a, \perp\}$

Initial state: q_0

Initial stack symbol: $\perp$

Final states: $\{q_2\}$

and the transition relation is

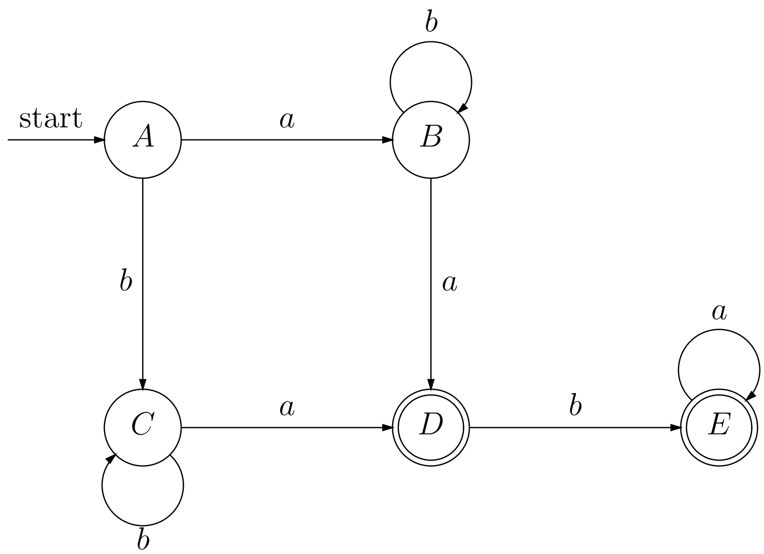

Figure 14: M_{14}

$$\delta = \{ \begin{array}{l} ((q_0, a, \perp), (q_0, a\perp)), \\ ((q_0, a, a), (q_0, aa)), \\ ((q_0, b, a), (q_1, \epsilon)), \\ ((q_1, b, a), (q_1, \epsilon)), \\ ((q_1, \epsilon, \perp), (q_2, \epsilon)) \quad \} \end{array}$$

Find the configurations that describe the actions of the automaton when the following strings are used as input. For each string, also state whether M accepts it by 1) final state and 2) empty stack.

a) aa

b) $aabba$

c) $aaabbb$

Is M a deterministic pushdown automaton? In other words, is its next configuration relation a (partial) function?

6.5 Let M be a PDA where

States: $\{q_0, q_1, q_2\}$

Input alphabet: $\{a, b, c, (,), +, -, \cdot, /\}$

Stack alphabet: $\{a, b, c, (,), +, -, \cdot, /, E, F, T, \perp\}$

Initial state: q_0

Initial stack symbol: $\perp$

Final states: $\{q_2\}$

and the transition relation δ consists of the following transitions

$((q_0, \epsilon, \bot), (q_1, E\bot))$
$((q_1, \epsilon, E), (q_1, T))$
$((q_1, \epsilon, E), (q_1, E+T))$
$((q_1, \epsilon, E), (q_1, E-T))$
$((q_1, \epsilon, T), (q_1, F))$
$((q_1, \epsilon, T), (q_1, T \cdot F))$
$((q_1, \epsilon, T), (q_1, T/F))$
$((q_1, \epsilon, F), (q_1, a))$
$((q_1, \epsilon, F), (q_1, b))$
$((q_1, \epsilon, F), (q_1, c))$
$((q_1, \epsilon, F), (q_1, (E)))$
$((q_1, a, a), (q_1, \epsilon))$
$((q_1, b, b), (q_1, \epsilon))$
$((q_1, c, c), (q_1, \epsilon))$
$((q_1, (, (), (q_1, \epsilon))$
$((q_1,),)), (q_1, \epsilon))$
$((q_1, +, +), (q_1, \epsilon))$
$((q_1, -, -), (q_1, \epsilon))$
$((q_1, \cdot, \cdot), (q_1, \epsilon))$
$((q_1, /, /), (q_1, \epsilon))$
$((q_1, \epsilon, \bot), (q_2, \epsilon))$

Is M a deterministic pushdown automaton? Find the configurations ([Hopcroft &Ullman] call them "instantaneous descriptions") that describe the actions of the automaton when the following strings are used as input. For each string, also state whether it belongs to $L(M)$.

a) $a \cdot b + c$

b) $a + a - b \cdot (a/b + b/c)$

6.6 Construct a DPDA (a PDA which is deterministic, conf. the explanation in 6.4) accepting the language $\{a^i b^j \mid 0 \leq i < j\}$.

6.7 For each of the following CFG's, construct a PDA which accepts the language generated by the CFG in question. (S is the start symbol, as usual.)

a) $S \to aAA$
$A \to aS \mid bS \mid a$

b) $S \to aA \mid aBB$
$A \to Ba \mid Sb$
$B \to bAS \mid \epsilon$

6.8 Show that the following languages are not context-free.

a) $L_1 = \{\, a^j b^k a^l \mid 0 < j < k < l \,\}$

b) $L_2 = \{\, w \in \{a, b, c\}^* \mid$ w has an equal number of a's, b's and c's $\}$

c) $L_3 = \{\, ww \mid w \in \{a, b\}^* \,\}$

6.1 a) $S \to 00A1$
$A \to \epsilon \mid 0A \mid 1A$

b) $S \to 101A010B$
$A \to \epsilon \mid 101A$
$B \to \epsilon \mid 010B$

c) $S \to A11B$
$A \to \epsilon \mid 11A \mid 010A$
$B \to \epsilon \mid 00B \mid 11B$

6.2 a) NFA$_\epsilon$ M_{32} in figure 32.

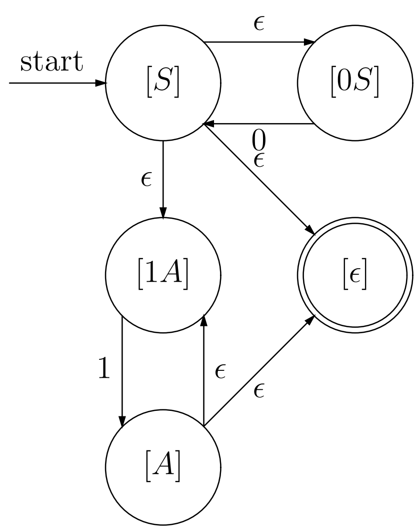

Figure 32: M_{32}

b) NFA$_\epsilon$ M_{33} in figure 33.

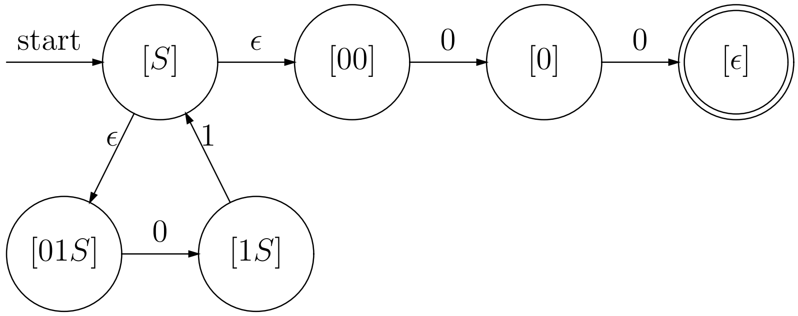

Figure 33: M_{33}

c) NFA$_\epsilon$ M_{34} in figure 34.

6.3 a) The principle: a nonterminal X generates a terminal string w iff w moves the automaton from state X to a final state.
In the grammar below, A is the start symbol.

$A \to 0B$
$B \to 1C \mid 1$
$C \to 0B$

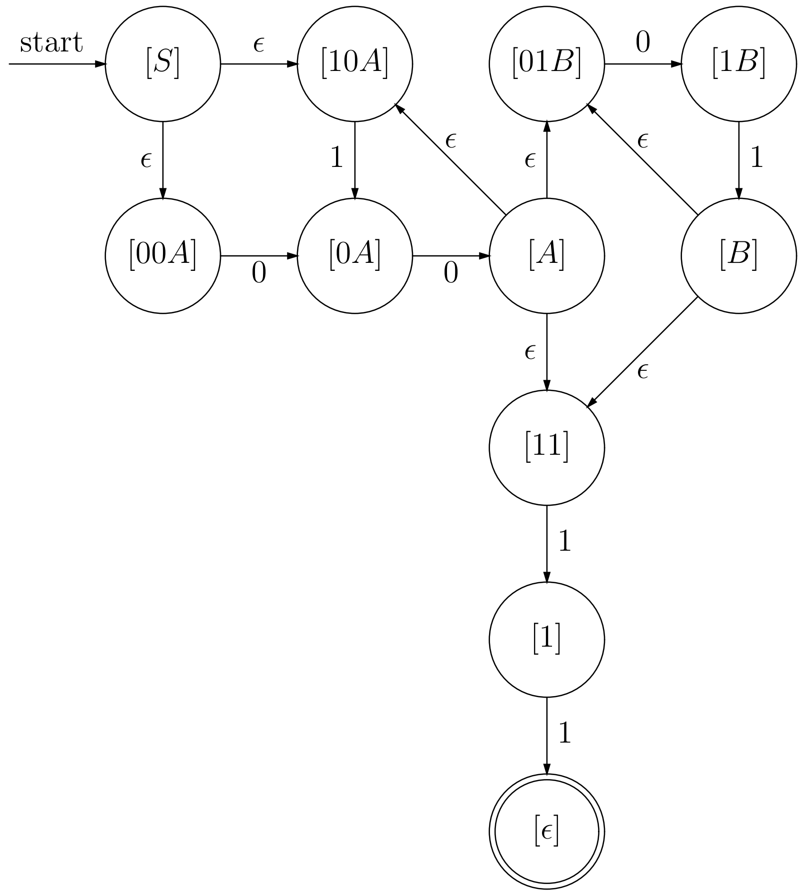

Figure 34: M_{34}

b) A is the start symbol

$A \to aB \mid bC$
$B \to aD \mid a \mid bB$
$C \to aD \mid a \mid bC$
$D \to bE \mid b$
$E \to aE \mid a$

6.6 Our automaton is $M = (\{q_0, q_1, q_2\}, \{a, b\}, \{a, \perp\}, \delta, q_0, \perp, \{q_2\})$, where δ is defined below. It accepts $\{a^i b^j \mid 0 \leq i < j\}$ by final state. The role of the states is described by

state	consumed input	stack	
q_0	a^i	$a^i\perp$	where $i \geq 0$
q_1	$a^i b^j$	$a^{i-j}\perp$	where $i \geq j > 0$
q_2	$a^i b^i b^k$	$\perp$	where $i \geq 0,\ k > 0$

$$\begin{array}{llll} \delta = \{ & ((q_0, a, \perp), (q_0, a\perp)), & ((q_1, b, a), (q_1, \epsilon)), & \\ & ((q_0, b, \perp), (q_2, \perp)), & ((q_1, b, \perp), (q_2, \perp)), & \\ & ((q_0, a, a), (q_0, aa)), & ((q_2, b, \perp), (q_2, \perp)), & \\ & ((q_0, b, a), (q_1, \epsilon)) & & \} \end{array}$$

6.7 The idea is to use the stack to simulate a leftmost derivation of the grammar. If the PDA has read w from the input, the stack contains $\gamma\bot$ and the state is q_1 then $S \Rightarrow^* w\gamma$.

a) $M = (\{q_0, q_1, q_2\}, \{a, b\}, \{a, b, S, A, \bot\}, \delta, q_0, \bot, \{q_2\})$, where

$$\begin{aligned}
\delta = \{ \; & ((q_0, \epsilon, \bot), (q_1, S\bot)), \\
& ((q_1, \epsilon, S), (q_1, aAA), \\
& ((q_1, \epsilon, A), (q_1, aS)), \\
& ((q_1, \epsilon, A), (q_1, bS)), \\
& ((q_1, \epsilon, A), (q_1, a)), \\
& ((q_1, a, a), (q_1, \epsilon)), \\
& ((q_1, b, b), (q_1, \epsilon)), \\
& ((q_1, \epsilon, \bot), (q_2, \epsilon)) \; \}.
\end{aligned}$$

b) $M = (\{q_0, q_1, q_2\}, \{a, b\}, \{a, b, S, A, B, \bot\}, \delta, q_0, \bot, \{q_2\})$, where

$$\begin{aligned}
\delta = \{ \; & ((q_0, \epsilon, \bot), (q_1, S\bot), && ((q_1, a, a), (q_1, \epsilon)), \\
& ((q_1, \epsilon, S), (q_1, aA)), && ((q_1, b, b), (q_1, \epsilon)), \\
& ((q_1, \epsilon, S), (q_1, aBB)), && ((q_1, \epsilon, \bot), (q_2, \epsilon)), \\
& ((q_1, \epsilon, A), (q_1, Ba)), \\
& ((q_1, \epsilon, A), (q_1, Sb)), \\
& ((q_1, \epsilon, B), (q_1, bAS)), \\
& ((q_1, \epsilon, B), (q_1, \epsilon)) \; \}
\end{aligned}$$

Both automata accept by final state.

6.8 a) Assume that L_1 is context-free. Then the pumping lemma holds. According to the lemma, there exists a number n such that if a string z, not shorter than n, is in L_1 (i.e. $|z| \geq n$, $z \in L_1$) then z can be split into five strings u, v, w, x, y:

$$z = uvwxy$$

such that

$$|vx| \geq 1, \qquad |vwx| \leq n, \qquad uv^iwx^iy \in L_1 \;\text{ for all } i \geq 0$$

We show that this leads to a contradiction. Take

$$z = a^n b^{n+1} a^{n+2} \in L_1.$$

Then there exist strings u, v, w, x, y satisfying the conditions above. We have two possibilities:

1. vwx does not overlap with the initial a^n. In other words, $u = a^n u'$ (for some u'). Take $i = 0$. Then $uv^0wx^0y = uwy = a^n b^k a^l$, for some k and l. The string $a^n b^k a^l$ is shorter than $a^n b^{n+1} a^{n+2}$ (as $|vx| \geq 1$), hence $k < n+1$ or $l < n+2$ (or both). In both cases $n < k < l$ is impossibe, so $uwy \notin L_1$ and we have a contradiction.

2. Otherwise vwx overlaps with the initial a^n. So it does not overlap with the final a^{n+2}, as $|vwx| \leq n$. In other words $y = y'a^{n+2}$, for some y'. Take $i = 2$. If uv^2wx^2y is not of the form $a^jb^ka^l$ then $uv^2wx^2y \notin L_1$, contradiction. If $uv^2wx^2y = a^jb^ka^l$ then $l = n+2$ but $j > n$ or $k > n+1$ (as $|vx| \geq 1$). Thus $j < k < l$ does not hold and $uv^2wx^2y \notin L_1$. Contradiction.

This completes the proof. As usually in proofs with pumping lemmas, choosing an appropriate string z was crucial. For instance, if $z = ab^na^{2n}$ then we may take $v = w = \epsilon$, $x = a$, $u = ab^n$ and and we do not obtain contradiction.

b) We know (cf. the book) that the language $M = \{\, a^ib^ic^i \mid i \geq 0 \,\}$ is not context-free. Notice that $L_2 \cap a^*b^*c^* = M$. Remember that the intersection of a context-free language with a regular language is context-free. So if L_2 were context-free then M would also be context-free. The latter is not true, so L_2 is not context-free.

(A proof using the pumping lemma is also possible; take for instance $z = a^nb^nc^n$).

c) To show that L_3 is not a context-free language, we show that the pumping lemma does not hold for L_3. To do this, for every number n we have to find a string $z \in L_3$, $|z| \geq n$ such that for every splitting of z into five pieces

$$z = uvwxy, \quad \text{where } |vx| \geq 1 \text{ and } |vwx| \leq n$$

some of the strings uv^iwx^iy $(i = 0, 1, \ldots)$ are not in L_3.

Let us try with

$$z = a^nb^na^nb^n \in L.$$

Consider an arbitrary splitting as above. If $|vx|$ is odd then uv^0wx^0y has an odd length and thus is not in L_3. So it remains to consider the case of $|vx|$ being even. Notice that $3n \leq |uv^0wx^0y| \leq 4n - 1$. We have three possibilities:

1. vwx is contained in the first half of z (so $y = y'a^nb^n$, for some y'). Then the last symbol of the first half of uv^0wx^0y is a (we removed some symbols from the first half of z and "the middle moved to the right"). Thus uv^0wx^0y is not in L_3 (as the last symbol of its second half is b).
2. vwx is contained in both halves of z. So it begins with a b and ends with an a. Thus v contains (at least one) b or x contains (at least one) a. Thus the first half of uv^0wx^0y has fewer b's than the second one[1] or the second half of uv^0wx^0y has fewer a's than the first one. Hence uv^0wx^0y is not in L_3.
3. vwx is contained in the second half of z (so $u = a^nb^nu'$, for some u'). This case is symmetric to case 1. The first symbol of the second half of uv^0wx^0y is b and $uv^0wx^0y \notin L_3$.

---

[1] As the halves of uv^0wx^0y are not shorter than $1.5n$, the first half begins with a^n and the second half ends with b^n.

Notice that in our proof we could not choose, for instance, $z = a^n b a^n b$. (Then there exists a splitting $z = uvwxy$ satisfying the conditions of the pumping lemma such that $uv^i wx^i y \in L_3$ for all $i \geq 0$).

For another proofs that L_3 is not context-free, see [Kozen, p. 154] and [Hopcroft&Ullman, p. 136].