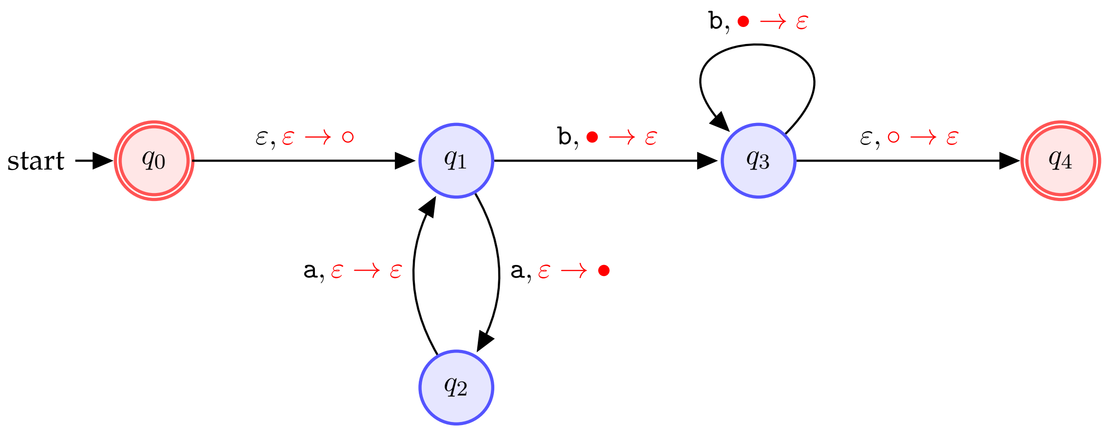
Basic

You may use JFLAP to help yourself work on these exercises.
You may wish to go through the tutorial sections: “Context-free Grammar” and “Pushdown Automata” available at http://www.jflap.org/modules/ (accessible through the left yellow navigation pane) or go through the relevant chapters in the JFLAP book (available as PDF with JFLAP in Moodle).

(1) Consider the following PDA



1) Produce the formal definition for the above NFA. This should consist of:

- The set of states $Q = \{\square, \square, \square, \square, \square\}$
- The input alphabet $\Sigma = \{\square, \square\}$
- The stack alphabet $\Gamma = \{\square, \square\}$
- The transition function, $\delta\colon Q \times \Sigma_\varepsilon \times \Gamma_\varepsilon \to 2^{Q\times\Gamma_\varepsilon}$, in table form

Σ_ε :	a			b			ε		
Γ_ε :	$\bullet$	$\circ$	ε	$\bullet$	$\circ$	ε	$\bullet$	$\circ$	ε
q_0									$\{(q_1, \circ)\}$
q_1			$\{(q_2, \bullet)\}$	$\{(q_3, \varepsilon)\}$					
q_2			$\square$						
q_3				$\{(q_3, \varepsilon)\}$				$\square$	
q_4									

The $\emptyset$ entries have been left blank to make the table easier to read.

- The start state $q_{\text{start}} = \square$
- The set of accept states $F = \{\square, \square\}$

2) Simulate the following strings: (For each step record: the state, the symbol just read and the stack contents)

aaab   aaaab   aab   aabb   aaaabb

3) Use **set notation** to describe the language recognized by this PDA.

$$\{ \ \mathtt{a}^{\square}\mathtt{b}^{\square} \ \mid \ n \geq \square \ \}$$

Basic

(2) For each of the Context-Free Grammars (CFGs) given below, give answers to the accompanying questions (together with short justifications where needed)

1)

$$\begin{aligned} A &\rightarrow \texttt{bb}A\texttt{b} \mid B \\ B &\rightarrow \texttt{a}B \mid \varepsilon \end{aligned}$$

Use the grammar to derive the following strings

$\texttt{bbab}$ $\texttt{bbb}$ $\texttt{a}^6$ $\texttt{b}^4\texttt{a}^3\texttt{b}^2$

2)

$$\begin{aligned} S &\rightarrow \texttt{a}A\texttt{bb} \mid \texttt{b}B\texttt{aa} \\ A &\rightarrow \texttt{a}A\texttt{bb} \mid \varepsilon \\ B &\rightarrow \texttt{b}B\texttt{aa} \mid \varepsilon \end{aligned}$$

Use the grammar to derive the following strings (where possible):

$\texttt{aabbbb}$ $\texttt{bbaaaa}$ $\texttt{aabb}$ $\texttt{baa}$

3) Here $\Sigma = \{\texttt{a}, +, \times, (,)\}$.

$$\begin{aligned} E &\rightarrow E + T \mid T \\ T &\rightarrow T \times F \mid F \\ F &\rightarrow (E) \mid \texttt{a} \end{aligned}$$

Give parse trees for each of the following strings

$\texttt{a}$ $\texttt{a} + \texttt{a}$ $\texttt{a} + \texttt{a} + \texttt{a}$ $(\texttt{a}) + (\texttt{a} + \texttt{a})$ $((\texttt{a}))$

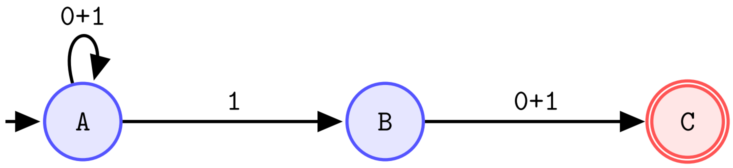
4) You are given the following CFG productions

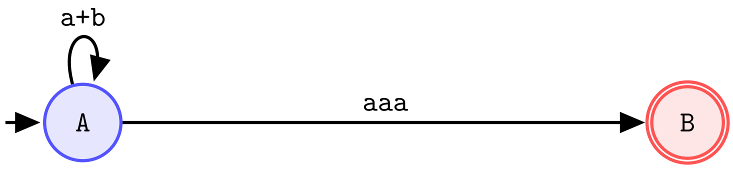
$$\begin{aligned} R &\rightarrow XRX \mid S \\ S &\rightarrow \texttt{a}T\texttt{b} \mid \texttt{b}T\texttt{a} \\ T &\rightarrow XTX \mid X \mid \varepsilon \\ X &\rightarrow \texttt{a} \mid \texttt{b} \end{aligned}$$

1. What are the variables (non-terminals)? $V = \{\square, \square, \square, \square\}$
2. What are the terminals? $\Sigma = \{\square, \square\}$
3. What is the start variable? $\square$
4. Give three strings in $L(G)$ $\square$, $\square$, $\square$
5. Give three strings not in $L(G)$ $\square$, $\square$, $\square$
6. True or False:
   (a) $T \rightarrow \texttt{aba}$
   (b) $T \xrightarrow{*} \texttt{aba}$
   (c) $T \rightarrow T$
   (d) $T \xrightarrow{*} T$
   (e) $XXX \xrightarrow{*} \texttt{aba}$
   (f) $X \xrightarrow{*} \texttt{aba}$
   (g) $T \xrightarrow{*} XX$
   (h) $T \xrightarrow{*} XXX$
   (i) $S \xrightarrow{*} \varepsilon$

Notation:
$\rightarrow$: in *one* step;
$\xrightarrow{*}$: in *zero or more* steps

(3) Convert the following GNFAs into **regular grammars**.





(4) Design a PDA and a CFG for the following language over $\Sigma = \{\mathtt{a}, \mathtt{b}\}$

$$L = \{w \mid w = (\mathtt{ab})^n \text{ or } w = \mathtt{a}^{4n}\mathtt{b}^{3n} \text{ for } n \geq 0\}.$$

Do this in two steps:

1) Explain the idea used, i.e. *how does the stack help you?*
2) Design a state diagram for the PDA.
3) Design a CFG.

(5) A string w is a *palindrome* if $w = w^R$, where w^R is formed by writing the symbols of w in reverse order, e.g. if $w = \mathtt{011}$ then $w^R = \mathtt{110}$.

Design PDAs and CFGs for each of the following languages

1) $\{w \mid w = \mathtt{b}^n\mathtt{ab}^n, \quad n \geq 0\}$
2) $\{w\mathtt{c}w^R \mid w \in \{\mathtt{a}, \mathtt{b}\}^*\}$ (so it is defined over the alphabet $\{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$)
3) $\{ww^R \mid w \in \{\mathtt{a}, \mathtt{b}\}^*\}$
4) The language of palindromes over $\{\mathtt{a}, \mathtt{b}\}$
5) The language of palindromes over $\{\mathtt{a}, \mathtt{b}\}$ whose length is a multiple of 3

   Hint: Consider the even and odd length cases first.

(6) **(Ambiguity)** Sometimes a grammar can generate the same string in several different ways, with several different parse trees, and likely several different meanings. If this happens, we say that the string is derived *ambiguously* in that grammar, which is then qualified as being an **ambiguous** grammar.

Consider the CFG

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

Derive the string $a + a \times a$ in two different ways using parse trees, and explain their (different) meanings.

Now note that the following alternative CFG is *not* ambiguous:

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

What is the parse tree for the previous example string $(a + a \times a)$?
What is the parse tree for $(a + a) \times a$?

Intermediate

(1) Design CFGs generating the following languages.

1) The language of all strings over $\{\mathtt{a}, \mathtt{b}\}$ with a single symbol '$\mathtt{b}$' located *exactly in the middle* of the string.

$$\{\mathtt{b}, \mathtt{aba}, \mathtt{abb}, \mathtt{bba}, \mathtt{bbb}, \mathtt{aabaa}, \ldots\}$$

2) The language of strings over $\{\mathtt{a}, \mathtt{b}\}$ containing an equal number of $\mathtt{a}$'s and $\mathtt{b}$'s.

3) The language of strings with twice as many $\mathtt{a}$'s as $\mathtt{b}$'s.

4) $\{\mathtt{a}^i\mathtt{b}^j \mid i, j \geq 0 \text{ and } i \geq j\}$

5) $\{\mathtt{a}^i\mathtt{b}^j \mid i, j \geq 0 \text{ and } i \neq j\}$ (Complement of the language $\{\mathtt{a}^n\mathtt{b}^n \mid n \geq 0\}$)

6) The language of strings over $\{\mathtt{a}, \mathtt{b}\}$ containing more $\mathtt{a}$'s than $\mathtt{b}$'s. (e.g. $\mathtt{abaab}$)

7) $\{w\#x \mid w, x \in \{\mathtt{0}, \mathtt{1}\}^* \text{ and } w^R \text{ is a substring of } x\}$

8) $\{x_1\#x_2\#\cdots\#x_k \mid k \geq 1, \text{ each } x_i \in \{\mathtt{a}, \mathtt{b}\}^*, \text{ and for some } i \text{ and } j,\ x_i = x_j^R\}$

Give informal descriptions of PDAs for the above languages. (How would you use the stack?)

(2) Let $\Sigma = \{\mathtt{a}, \mathtt{b}\}$ and let B be the language of strings that contain at least one $\mathtt{b}$ in their second half. In other words, $B = \{uv \mid u \in \Sigma^*, v \in \Sigma^*\mathtt{b}\Sigma^* \text{ and } |v| \leq |u|\}$.

1) Give a PDA that recognizes B.

2) Give a CFG that generates B.

(3) Let

$$\begin{aligned} C &= \{x\#y \mid x, y \in \{\mathtt{0}, \mathtt{1}\}^* \text{ and } x \neq y\} \\ D &= \{x\#y \mid x, y \in \{\mathtt{0}, \mathtt{1}\}^* \text{ and } |x| = |y| \text{ but } x \neq y\} \end{aligned}$$

Show that C and D are both CFLs by producing PDAs or CFGs for them.

(4) Give a **counter example** to show that the following construction fails to prove that the class of context-free languages is closed under star.

> Let A be a CFL that is generated by the CFG $G = (V, \Sigma, R, S)$.
> Add the new rule $S \to SS$ and call the resulting grammar G'.
> This grammar is supposed to generate A^*.

*Note: the class of **context-free languages** is actually **closed under the regular operations** (union, concatenation, and star) but the above argument fails to prove closure under star. What is missing?*

Extend your class for simulating NFAs from lab 2 to simulate PDAs.

Advanced