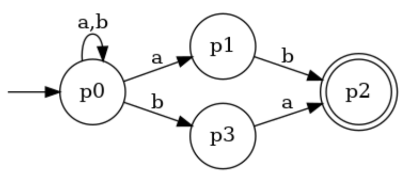
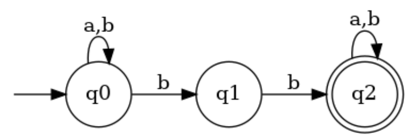
This homework covers sections 3.7, 4.1, and 4.2, with a little left over from section 3.6. It is due in class Wednesday, April 10. Hand in a hardcopy of your solutions.

While you may discuss problems with other students, you should always make the first attempt on a problem yourself and **you must write up your own solutions in your own words**. *You may not collaboratively write solutions or copy a solution that one person in the group writes up.*

1. Use the construction outlined in the proof of Theorem 3.5 to construct a machine that accepts the intersection of the languages accepted by the two NFAs shown. (Hint: remember that the construction outlined in the proof of Theorem 3.5 is for two *DFAs*.)





2. Use the Pumping Lemma (Theorem 3.6) to prove that the following languages are not regular.

   Remember the pattern for the proof discussed in class, which can be written:

   > Suppose L is regular. Then, by the Pumping Lemma, there is an integer N such that for any string $w \in L$ with $|w| \geq K$, $w = xyz$ where $|xy| \leq N$, $|y| \geq 1$, and $xy^nz \in L$ for all natural numbers n. But let $w =$ _____, which is in L with $|w| \geq N$. Write w as xyz. Because $|xy| \leq N$, y must be of the form _____. But then $xy^nz \notin L$ for $n =$_____ because ____________________.

   Fill in the blanks.

   (a) $L = \{ \ a^n b^n c^n \mid n \in \mathbb{N} \ \}$

   (b) $L = \{ \ w \in \{a, b\}^* \mid n_a(w) < n_b(w) \ \}$, where $n_\sigma(w)$ means the number of σs in w

3. Consider the context-free grammar shown on the right.

$$\begin{aligned} S &\longrightarrow TR \\ T &\longrightarrow aTb \\ T &\longrightarrow \epsilon \\ R &\longrightarrow cRd \\ R &\longrightarrow c \end{aligned}$$

   (a) Write a derivation for the string *aabbc* using this grammar.

   (b) Write a derivation for the string *abcccdd* using this grammar.

   (c) Find the language generated by this grammar. Briefly justify your answer.

4. For each of the following languages, give a context-free grammar that generates the language. Also explain in words why your grammar works.

   Hint for part (d): keep in mind how to generate equal numbers of things, and think about which pairs of characters can match up with each other.

   a) $\{a^n b^m \mid n \neq m\}$
   b) $\{a^n b^m c^k \mid m > n + k\}$
   c) $\{a^n b^m c^k d^l \mid m = k \text{ and } n = l\}$
   d) $\{a^n b^m c^k d^l \mid n + m = k + l\}$

5. Given the following BNF grammar for a (very small) subset of Java syntax, write down six things generated by this grammar. Your examples should demonstrate all of the possibilities represented in the rules — make sure to use each of the different rules and variations in at least one derivation.

$$\begin{aligned} \langle name \rangle &::= \langle object\_ref \rangle \text{ ["." } \langle identifier \rangle \text{]} \\ \langle object\_ref \rangle &::= \langle identifier \rangle \mid \langle method\_call \rangle \\ \langle method\_call \rangle &::= \langle identifier \rangle \text{ "(" } \langle name \rangle \text{ ["," } \langle name \rangle \text{]} \ldots \text{ ")"} \\ \langle identifier \rangle &::= \text{"}a\text{"} \mid \text{"}b\text{"} \mid \text{"}c\text{"} \mid \text{"}x\text{"} \mid \text{"}y\text{"} \mid \text{"}z\text{"} \end{aligned}$$

6. Write a BNF grammar for numeric literals in Java. Numeric literals include integers (e.g. 0, 1035, -47, +2) and floating point numbers (e.g. 17.3, 0.73, .50, 23.1e67, -1.34E-12, +0.2, 100E+100). Note that the only numbers that can start with 0 are the integer 0 and a floating point number between -1 and 1.