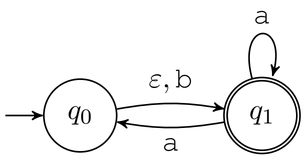
Problem 1 (25 points)

Let $\Sigma = \{\texttt{a}, \texttt{b}\}$. Consider the language L of following NFA M:



(a) (14 points) Give a minimal DFA for the language L. Prove that your DFA is minimal (for every pair of states, give a string to distinguish them).

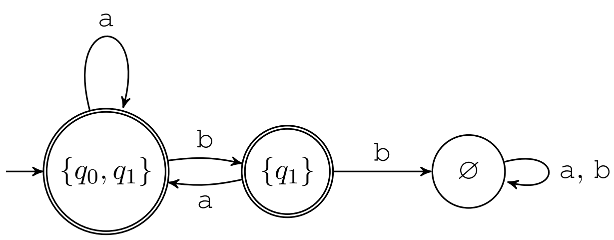
If your DFA is not minimal you will get partial credits.

(b) (6 points) Write a regular expression for L. *Hint: There is a simple solution.*

(c) (5 points) Write a regular expression for the complement of L.

Solution:

(a) We convert the NFA into a DFA using the subset construction.



6 points for writing the correct DFA for the language L. 2 points for correct minimization.

$\{q_0, q_1\}$ is distinguishable from $\{q_1\}$ by b.
$\{q_0, q_1\}$ is distinguishable from $\varnothing$ by ε (or a, or b).
$\{q_1\}$ is distinguishable from $\varnothing$ by ε (or a).
2 points each for distinguishing the three pairs of states.

(b) L represents strings without two consecutive b's. Regular expressions like `(a+ba)*(`ε`+b)`, `(`ε`+b)(a+ab)*`, $\texttt{a}^*\texttt{b}(\texttt{aa}^*\texttt{b})^*\texttt{a}^*$ and some other forms are correct. Incorrect regular expression without any procedure would get 0 points. 2 points for some correct procedure.

(c) `(a+b)*bb(a+b)*`. The complement of L represents strings with two consecutive b's. Other regular expressions are correct if it represents strings with 2 consecutive b's.

Problem 2 (20 points)

Consider the language

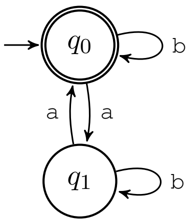
$$L = \{w \in \{\mathtt{a}, \mathtt{b}\}^* \mid w \text{ contains an even number of } \mathtt{a}\text{'s and ends with the symbol } \mathtt{b}\}.$$

(a) (10 points) Give a DFA for the language L. Briefly explain how you obtain your DFA (insufficient explanation will get no points).

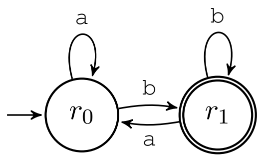
(b) (10 points) Give a context-free grammar for L. Briefly explain how your context-free grammar works (insufficient explanation will get no points).

Solution:

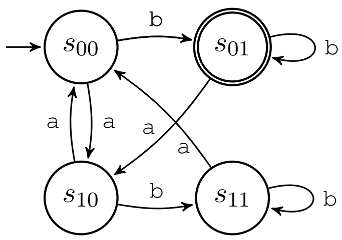
(a) DFA for strings with an even number of a's:



DFA for strings ending in b:



DFA for the intersection of the two languages:



(b) Any string in L must be of the form $w\mathtt{b}$ for some string w having an even number of a's (call this language L'). There are three cases for such a string w with an even number of a's:

(i) w begins with a b: $w = \mathtt{b}t$ for some $t \in L'$;

(ii) w begins with an a: $w = \mathtt{a}u\mathtt{a}v$ for some $u, v \in L'$, because the a at the beginning must match some other a, such that the substring between the two a's and the remaining part after that other a must themselves have an even number of a's;

(iii) w is the empty string.

This gives us the following context-free grammar:

$$S \to T\mathtt{b}$$
$$T \to \mathtt{b}T \mid \mathtt{a}T\mathtt{a}T \mid \varepsilon$$

Problem 3 (20 points)

Consider the language

$$L = \{w \in \{\texttt{a}, \texttt{b}\}^* \mid w \text{ contains no more } \texttt{a}\text{'s than } \texttt{b}\text{'s}\}.$$

For example, ab and bab are in L, but baa is not.

(a) (10 points) Prove that L is irregular.

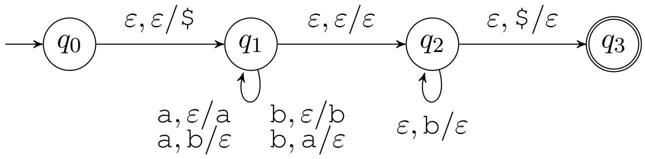
(b) (10 points) Give a pushdown automaton for the language L. Briefly explain how your pushdown automaton works (insufficient explanation will get no points).

Solution:

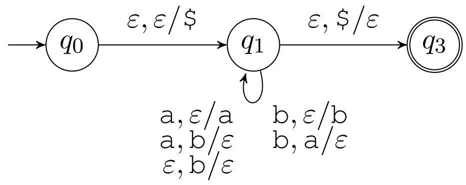
(a) We prove irregularity of L using pairwise distinguishable strings. We claim that $\{\texttt{a}^i \mid i \geqslant 0\} = \{\varepsilon, \texttt{a}, \texttt{aa}, \texttt{aaa}, \dots\}$ is an infinite list of strings that are pairwise distinguishable by L.

Indeed, take two different strings $x = \texttt{a}^i$ and $y = \texttt{a}^j$ from the list, and assume $i < j$. If we take the extension string $z = \texttt{b}^i$, we have $xz \in L$ but $yz \notin L$.

(b) Our pushdown automaton is a modification of the one on the last page of Lecture 10. In state q_1, we keep track of the excess of a's over b's (or the other way round). In state q_2, we make sure our stack contains only b's but no a's (hence an excess of b's over a's) before going to the accepting state.



Alternatively, we may combine states q_1 and q_2 together:



Problem 4 (35 points)

Consider language

$$L = \{w \in \{\mathtt{a}, \mathtt{b}\}^* \mid w \text{ contains the same number of } \mathtt{a}\text{'s and } \mathtt{b}\text{'s}\}.$$

The language L may be described by the following context-free grammar G:

$$S \to \mathtt{a}S\mathtt{b}S \mid \mathtt{b}S\mathtt{a}S \mid \varepsilon$$

(a) (7 points) Show that the grammar G is ambiguous. *Hint: Some string of length at most four will help you.*

(b) (8 points) Convert the grammar to Chomsky Normal Form.

Solution:

(a) The string `abab` has two different parse trees:

Note that showing ambiguity requires demonstrating two different *parse trees* (or different *leftmost derivations*). It is not sufficient to just demonstrate two different *derivations* (which may have the same parse tree).

(b) First, add a new start variable S_0 and the rule $S_0 \to S$ (required because the original start variable S appears on the right of some rule, and S can generate ε). Then eliminate ε-production and unit productions. Finally we break any long production rule into shorter ones:

$$\begin{aligned}
S_0 &\to XY \mid YX \mid AY \mid YA \mid BX \mid XB \mid AB \mid BA \mid \varepsilon \\
S &\to XY \mid YX \mid AY \mid YA \mid BX \mid XB \mid AB \mid BA \\
X &\to AS \\
Y &\to BS \\
A &\to \mathtt{a} \\
B &\to \mathtt{b}
\end{aligned}$$

A shorter, alternative solution:

$$\begin{aligned}
S_0 &\to XY \mid YX \mid \varepsilon \\
S &\to XY \mid YX \\
X &\to AS \mid \mathtt{a} \\
Y &\to BS \mid \mathtt{b} \\
A &\to \mathtt{a} \\
B &\to \mathtt{b}
\end{aligned}$$

3-5 points for the correct intermediate grammar that is very far from Chomsky Normal Form. -1-3 for a grammar reasonably close to Chomsky Normal Form, but has some mistakes. For incomplete answers, -1 for each missing situation.

Problem 4 (continued)

(c) (10 points) Using the grammar from part (b), apply the Cocke–Younger–Kasami on input `abba`. Show the table of partial derivations. Draw a parse tree derived by the algorithm.

(d) (10 points) Consider the context-free language

$$L' = \{w \in \{\texttt{a}, \texttt{b}\}^* \mid w \text{ is a palindrome}\}.$$

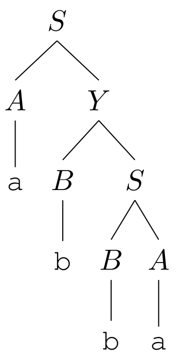
Recall that a string w is a palindrome if it reads the same forward and backward.

Show that $L \cap L'$ is not context-free. (Note: This shows that context-free languages are not closed under intersection.)

Solution:

(c)

$S_0 \mid S$			
–	Y		
$S_0 \mid S$	–	$S_0 \mid S$	
A	B	B	A
a	b	b	a



Students' answer in part (c) has to be consistent with their grammar in part (b). If the answer in part (b) is not correct (not in the right form), some points (1-5) may be deducted for part (c). 5 points for the correct table. 5 points for the correct parse tree.

(d) We show $S = L \cap L'$ is not context-free by the pumping lemma. Suppose towards a contradiction that S is context-free. Let p be the pumping length. Define $s = \texttt{a}^p\texttt{b}^p\texttt{b}^p\texttt{a}^p$ so that $s \in S$. The pumping lemma then splits s as $s = uvwxy$ such that $|vwx| \leqslant q$, $|vx| > 0$, and $uv^iwx^iy \in L$ for every $i \geqslant 0$. We show this is impossible by considering two cases:

(i) Case 1: v and x consist only of b's, then uv^0wx^0y has fewer b's than a's. Therefore $uv^0wx^0y \notin S$.

(ii) Case 2: v or x contains an a.

Note that v and x cannot simultaneously contain a's from the leftmost block of a's and the rightmost block of a's, since $|vwx| \leqslant p$.

Let's say v or x contains an a from the leftmost block (the other case is similar). then uv^0wx^0y removes some a's on the leftmost block of a's without affecting the rightmost block of a's. Therefore uv^0wx^0y is not a palindrome and is not in S.

This covers all cases. Therefore $L \cap L'$ is not context-free.