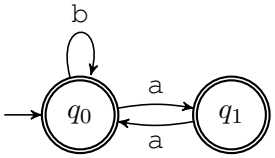
## Problem 1 (20 points)

(a) (10 points) Convert the following NFA into a DFA.



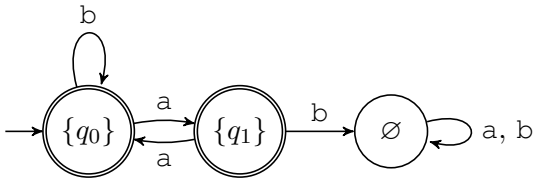

(b) (10 points) Give a DFA for the following language:

$$L = \{w \in \{\mathtt{a}, \mathtt{b}\}^* \mid w \text{ begins with } \mathtt{a} \text{ or ends with } \mathtt{b}\}$$

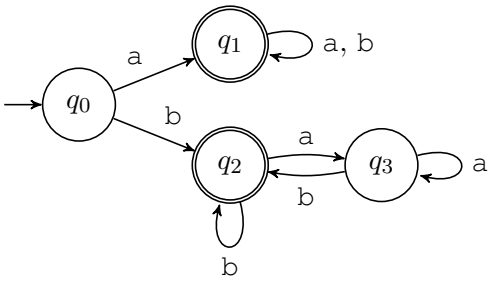
Briefly explain how your DFA works (insufficient explanation will get no points).

**Solution:**

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



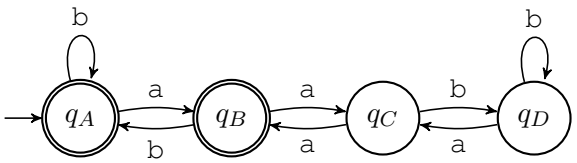

(b) In the following DFA, if the first symbol is a, it goes to the $q_1$ that always accept. If the first symbol is b, it goes to the lower branch consisting of $q_2$ and $q_3$, which accepts if and only if the last symbol is a b.



**Problem 2 (25 points)** The language

$$L = \{w \in \{\mathtt{a}, \mathtt{b}\}^* \mid w \text{ has an even number of occurrences of the substring } \mathtt{aa}\}$$

has a DFA




(a) (6 points) Prove that the above DFA is minimal: For every pair of distinct states, write down a string (in the upper triangular part of the following table) to distinguish them.

(b) (9 points) Suppose $L'$ is the language of the regular expression $\mathtt{a}^*\mathtt{b}^*$. Write a regular expression for $L \cap L'$. Briefly explain how your expression works (insufficient explanation will get no points).

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

**Solution:**

(a)

| | $q_A$ | $q_B$ | $q_C$ | $q_D$ |
|---|---|---|---|---|
| $q_A$ | $\times$ | $\mathtt{a}$ | $\varepsilon$ | $\varepsilon$ |
| $q_B$ | $\times$ | $\times$ | $\varepsilon$ | $\varepsilon$ |
| $q_C$ | $\times$ | $\times$ | $\times$ | $\mathtt{a}$ |
| $q_D$ | $\times$ | $\times$ | $\times$ | $\times$ |

Other solutions are possible.

(b) $(\varepsilon+\mathtt{a}(\mathtt{aa})^*)\mathtt{b}^*$

A string in $L \cap L'$ consists of a sequence of a's that contains an even number of aa (hence itself the empty string or an odd number of a's), followed by a sequence of b's.

(c) Solution 1:

$$\begin{aligned} A &\to \mathtt{a}B \mid \mathtt{b}A \mid \varepsilon \\ B &\to \mathtt{a}C \mid \mathtt{b}A \mid \varepsilon \\ C &\to \mathtt{a}B \mid \mathtt{b}D \\ D &\to \mathtt{a}C \mid \mathtt{b}D \end{aligned}$$

Solution 2:

$$\begin{aligned} S &\to A \mid B \\ A &\to A\mathtt{b} \mid B\mathtt{b} \mid \varepsilon \\ B &\to A\mathtt{a} \mid C\mathtt{a} \\ C &\to B\mathtt{a} \mid D\mathtt{a} \\ D &\to C\mathtt{b} \mid D\mathtt{b} \end{aligned}$$

Both CFGs simulate the given DFA. In Solution 1, if a string $w$ stops at state $q_X$ in the DFA, then the CFG yields the string $wX$, and vice versa (for example, the string aab stops at $q_D$, so the CFG yields the string aab$D$). Now a state is accepting if and only if its corresponding variable in the CFG can become the empty string, so the CFG precisely generates all strings that stop at an accepting state.

Solution 2 is similar, except that it yields a string from right to left.

**Problem 3 (20 points)** Consider the language

$$L = \{w \in \{\mathtt{a}, \mathtt{b}\}^* \mid w \text{ has exactly twice as many } \mathtt{a}\text{'s as } \mathtt{b}\text{'s}\}.$$

For example, $\mathtt{aba} \in L$ and $\varepsilon \in L$, but $\mathtt{ab} \notin L$ and $\mathtt{abaa} \notin L$.

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

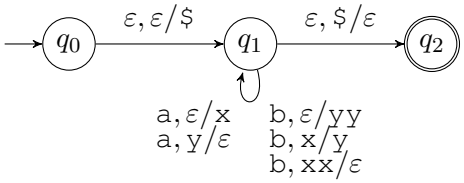
(b) (10 points) Give a pushdown automaton for $L$. Briefly explain how your pushdown automaton works (insufficient explanation will get no points).

**Solution:**

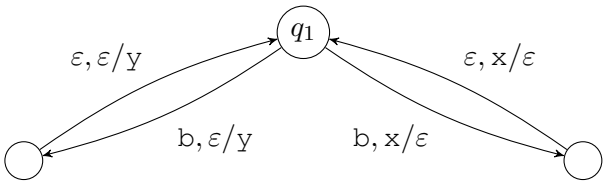
(a) We prove irregularity of $L$ using pairwise distinguishable strings. We claim that $\{\mathtt{a}^{2i} \mid i \geqslant 0\} = \{\varepsilon, \mathtt{aa}, \mathtt{aaaa}, \dots\}$ is an infinite list of strings that are pairwise distinguishable by $L$.

Indeed, take two different strings $x = \mathtt{a}^{2i}$ and $y = \mathtt{a}^{2j}$ from the list ($i \neq j$). If we take the extension string $z = \mathtt{b}^i$, 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. We keep track of $\#\mathtt{a} - 2(\#\mathtt{b})$ on the stack. If this number is positive and equals $n$, we keep $n$ many $\mathtt{x}$'s on the stack; if this number is negative and equals $-n$, we keep $n$ many $\mathtt{y}$'s (in at least one of the computational paths). An invariant that holds in all computational paths is $\#\mathtt{a} - 2(\#\mathtt{b}) = \#\mathtt{x} - \#\mathtt{y}$.




In the above, the transitions $\mathtt{b}, \varepsilon/\mathtt{yy}$ and $\mathtt{b}, \mathtt{xx}/\varepsilon$ are both shorthands. $\mathtt{b}, \varepsilon/\mathtt{yy}$ corresponds to the two transitions on the left of the following diagram, while $\mathtt{b}, \mathtt{xx}/\varepsilon$ corresponds to the two transitions on the right.



**Problem 4 (35 points)** The language

$$L = \{w \in \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}^* \mid \ w \text{ has no more } \mathtt{a}\text{'s than } \mathtt{b}\text{'s}\}$$

has a context-free grammar $G$:

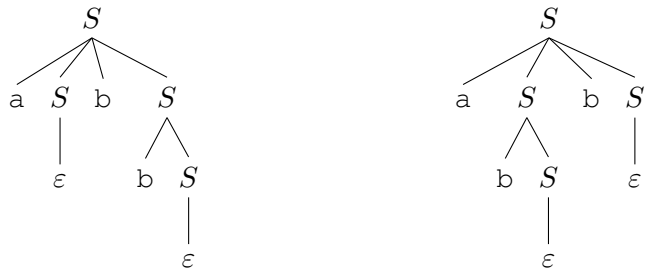
$$S \to \mathtt{a}S\mathtt{b}S \mid \mathtt{b}S\mathtt{a}S \mid \mathtt{b}S \mid \mathtt{c}S \mid \varepsilon$$

(a) (7 points) Show that $G$ is ambiguous. *Hint: Some string of length three over* $\{\mathtt{a}, \mathtt{b}\}$ *will help you.*

(b) (8 points) Convert $G$ to Chomsky Normal Form.

**Solution:**

(a) The string abb 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 $T$ and the rule $T \to S$ (required because the original start variable $S$ appears on the right of some rule, and $S$ can generate $\varepsilon$). Then eliminate $\varepsilon$-production and unit productions. Finally we break any long production rule into shorter ones:

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

A shorter, alternative solution:

$$\begin{aligned}
T &\to XY \mid YX \mid BS \mid CS \mid \\
&\quad \mathtt{b} \mid \mathtt{c} \mid \varepsilon \\
S &\to XY \mid YX \mid BS \mid CS \mid \\
&\quad \mathtt{b} \mid \mathtt{c} \\
X &\to AS \mid \mathtt{a} \\
Y &\to BS \mid \mathtt{b} \\
A &\to \mathtt{a} \\
B &\to \mathtt{b} \\
C &\to \mathtt{c}
\end{aligned}$$

**Problem 4 (continued)**

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

(d) (10 points) Consider the language

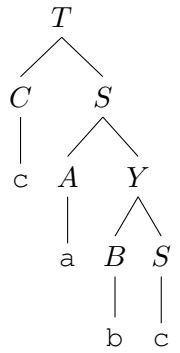
$$L' = \{w \in \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}^* \mid w \text{ has no more } \mathtt{b}\text{'s than } \mathtt{c}\text{'s}\}.$$

Show that $L \cap L'$ is not context-free.

**Solution:**

(c)

| | | | |
|---|---|---|---|
| $T\|S$ | | | |
| $T\|S$ | $T\|S\|X$ | | |
| – | $T\|S\|X$ | $T\|S\|Y$ | |
| $T\|S\|C$ | $A$ | $T\|S\|B$ | $T\|S\|C$ |
| c | a | b | c |




Students' answer in part (c) has to be consistent with their grammar in part (b).

(d) We show $K = L \cap L'$ is not context-free by the pumping lemma. Suppose towards a contradiction that $K$ is context-free. Let $p$ be the pumping length. Define $s = \mathtt{a}^p\mathtt{b}^p\mathtt{c}^p$ so that $s \in K$. 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 $x$ consists of `a`'s and/or `b`'s (but not `c`'s), then $uv^2wx^2y$ has more `a`'s or `b`'s than `c`'s. Therefore $uv^2wx^2y \notin K$.

(ii) Case 2: $v$ and $x$ consists of `b`'s and/or `c`'s (but not `a`'s), then $uv^0wx^0y$ has fewer `b`'s or `c`'s than `a`'s. Therefore $uv^0wx^0y \notin K$.

This covers all cases (it's impossible for $v$ and $x$ to contain both `a`'s and `c`'s at the same time, due to $|vwx| \leqslant q$). Therefore $L \cap L'$ is not context-free.

Note: Many students claimed that if $vwx$ contains `a` and `b`, then $uv^2wx^2y$ contains more `b`'s than `c`'s. This is not necessarily true, when $x = \varepsilon$ and $v$ contains only `a`'s (while $w$ contains some `b`).