In the following problems, we will be always working on the alphabet $\Sigma = \{0, 1\}$.

1. State whether each of the following statements is **true**, or **false**[1]. Provide a justification for your answer. Formal proofs are not required.

   (a) (7 points) If L is a context-free language, then $L.L$ is a context-free language.

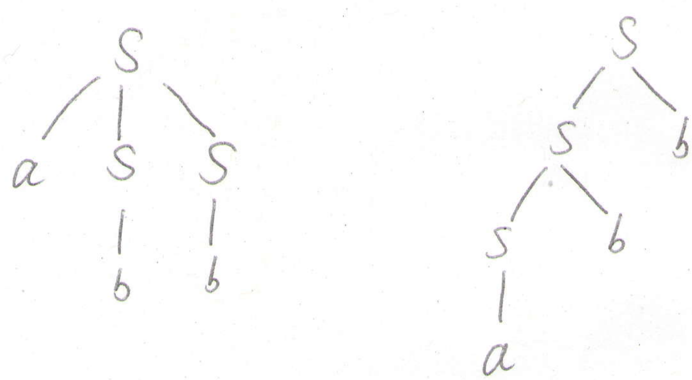
   True. The set of context-free languages is closed under concatenation.

   (b) (7 points) Consider the grammar $G = (\{S\}, \{a, b\}, R, S)$, where R has the rules

   $$S \to aSS \mid Sb \mid a \mid b$$

   Then G generates the string abb unambiguously.

   False. There are two parse trees for abb in G:



[1]but *not* both!

→

(c) (8 points) The language $\{a^m b^n c^n d^m \mid m, n \geq 0\}$ over $\Sigma = \{a, b, c, d\}$ is not context free.

False. Here is a CFG for this language:

$G = (\{S, T\}, \{a, b, c, d\}, R, S)$ where R is

$S \to aSd \mid T$

$T \to bTc \mid \epsilon$

(d) (8 points) Suppose L is a context-free language generated by the context-free grammar G. Let S be the start variable of G. Now we modify G by adding the rule $S \to SS$. Then this modified grammar, denoted by G', generates the language $L' \triangleq \{ww \mid w \in L\}$.

False. Here is a counter example:

$G = (\{S\}, \{a\}, R, S)$ where R is

$S \to a$.

Then $L(G) = \{a\} = L$

G' has the rules

$S \to SS \mid a$

So $L(G') = \{a^i \mid i \geq 1\} \neq L'$.

→

2. Consider the following language over $\Sigma = \{a, b\}$.

$L = \{w \mid w$ has an odd length, and the first, middle and last symbols of w are identical$\}$.

(a) (15 points) Give a CFG that generates L. Formal proof of correctness is not required, but you should justify your construction.

$$S \to aAa \mid bBb \mid C$$

$$A \to CAC \mid a$$

$$B \to CBC \mid b$$

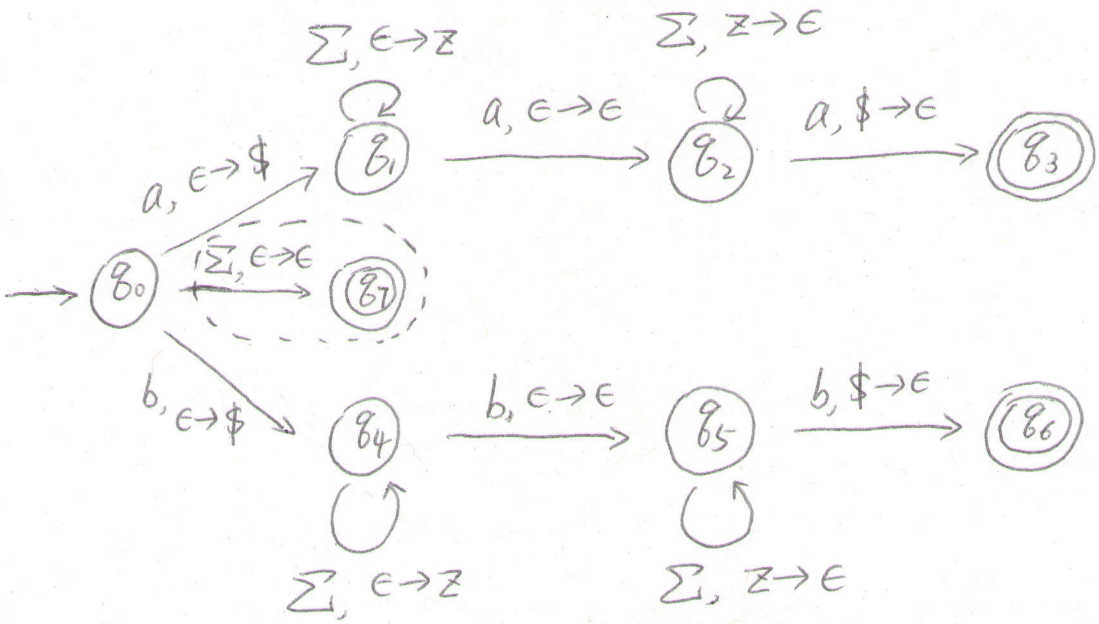
$$C \to a \mid b$$

Explanation: The $\underbrace{\text{strings in}}_{\text{(of length} \geq 3)}$ L are of the form $a S_1 a S_2 a$ or $b S_1 b S_2 b$ where $|S_1| = |S_2|$. We use variables A, B to generate strings of the form $S_1 a S_2$, $S_1 b S_2$ respectively. Then we append a or b to the two sides of A or B.

(The rule $S \to C$ is optional, depending on whether one considers $a, b \in L$ or not.)

→

(b) (15 points) Give a PDA that accepts L. Formal proof of correctness is not required, but you should justify your construction.



Explanation: This PDA checks that the input string starts and ends with the same symbol, and there is an identical symbol between them and it use the pushing / poping of the stack to check that the numbers of symbols between this symbol and the beginning and the end are equal.

(The state q_7 is optional, depending on whether one considers $a, b \in L$ or not.) →

3. (20 points) Show that the language $L = \{a^i b^j c^k \mid j = \max(i,k)\}$ over $\Sigma = \{a,b,c\}$ is not context-free.

Suppose L is context-free. Since it is infinite, let p be the constant in the pumping lemma.

Conside the string $a^p b^p c^p \in L$.

By the pumping lemma, there exist strings u, v, w, x, y, s.t. $a^p b^p c^p = uvwxy$, $|vwx| \le p$, $|vx| > 0$, and $uv^i w x^i y \in L$, $\forall i \ge 0$.

Now: ① If v or x contains two types of symbols, say, $v = a^i b^j$, $i,j \ge 1$, then uv^2wx^2y contains ba as substring, so $uv^2wx^2y \notin L$;

② o.w. v and x both contain only one type of symbol. So at least one of a, b, c is not contained by both v and x.

(2.1) if it is b, then uv^2wx^2y contains p b's, but more than p a's or c's. So $uv^2wx^2y \notin L$;

→

(2.2) O.W. vx contains b, but not a or c.
Then uwy contains p a's or c's, but
less than p b's. So $uwy \notin L$

In each case, we get a contradiction.
So L is not context-free.

4. (20 points) Given an arbitrary DFA $M = (Q, \Sigma, \delta, q_0, F)$, design a PDA $P = (Q', \Sigma, \Gamma, \delta', \delta'_0, F')$ such that $|Q'| \leq 3$ and P recognizes exactly $L(M)$ (i.e. the language recognized by M).

$P = (Q', \Sigma, \Gamma, \delta', q'_0, F')$ where

$Q' = \{q'_0, q'_1, q'_2\}$, $\Gamma = Q$, $F' = \{q'_2\}$,

$\delta'(q'_0, \epsilon, \epsilon) = \{(q'_1, q_0)\}$

$\delta'(q'_1, a, q) = \{(q'_1, \delta(q, a))\}$ $\forall\ a \in \Sigma$, $\forall\ q \in Q$

$\delta'(q'_1, \epsilon, q) = \{(q'_2, \epsilon)\}$ $\forall\ q \in F$.

Explanation: P simulates M by using the stack
to keep track of the state M is in when
processing any string. Specifically, at the beginning
it pushes q_0 onto the stack, and jumps to state
q'_1. Then, it stays in state q'_1, and keeps do
the following: it reads a symbol, pops out current
state of M, and push the new state of M to
the stack. Furthemore, when M is in a state
in F, P can also jump to state q'_2 which is
accepting. This acceptance will die out if →
a new symbol is seen.

5. (10 points) **BONUS QUESTION**
A language is *prefix-closed* if the prefix of any string in the language is also in the language. Show that every infinite prefix-closed context free language contains an infinite regular subset.
Hint: Use the pumping lemma.

Suppose L is infinite, prefix-closed and context-free.
Let p be the constant in the pumping lemma.
Since L is infinite, there exists $x \in L$, $|x| > p$.
By the pumping lemma, there exist strings u, v, w, y, z, s.t. $x = uvwyz$, $|vwy| \le p$, $|vy| > 0$, and $uv^i wy^i z \in L$, $\forall i \geq 0$.
Now: ① If $|v| > 0$, then, since L is prefix-closed, we have $uv^i \in L$, $\forall i \geq 0$. Clearly, $\{uv^i \mid i \geq 0\} \subseteq L$ is infinite and regular;
② O.W. $|v| = 0$, so $|y| > 0$. Thus, $uv^i w y^i z = uwy^i z \in L$. Since L is prefix-closed, $uwy^i \in L$. Then $\{uwy^i \mid i \geq 0\} \subseteq L$ is infinite and regular.

□