Theory of Computation, CSCI 438, Spring 2022
Exam 2, March 11 **Name __________________**

My phone number is **406-498-4884**. Feel free to call with any questions that you have.

There is one question per page.

Essay Questions

(20 pts.)

Definitions

1. Given a context-free grammar G = (V, Σ, R, S), give the definition of the language of that grammar. (10 pts.)

For G = (V, Σ, R, S),

$\mathcal{L}(G) = \{w \mid w \in \Sigma^* \wedge S \Rightarrow^* w\}$

Language Placement

2. Fill out the two columns, telling whether or not the language is regular and/or context free (10 pts.)

	Language	**Is the language regular?**	**Is the language context free?**
1	Language consisting of all strings on {0,1}* with the same number of 0's and 1's.	No	Yes
2	$L = \{w \mid n_a(w) \geq 2 \text{ and } n_b(w) > 2\}$ where $n_a(w)$ is the number of a's in w and $n_b(w)$ is the number of b's in w.	Yes	Yes
3	$L = \{a^n b^n a^n : n \geq 0\}$	No	No
4	$L = \{w \mid (n_a(w) - {}_b(w)) \bmod 3 = 1\}$	Yes	Yes

Problem Solving

3. Give a context-free grammar for the language L, on $\Sigma=\{0,1\}$, defined as:
$L = \{ 0^n 1^* 0^{2n} \mid n \geq 0 \}$ (10 pts.)

$S \rightarrow 0S00 \mid T$
$T \rightarrow 1T \mid \varepsilon$

4. Create a PDA for the language L, defined on {a,b,#}, as follows:

$L = \{x_1\# x_2\# \ldots \# x_k \mid \text{for } k \geq 2,\ x_1, x_2, \ldots, x_k \in \{a,b\}^* \text{and for some } i,\ x_i = x_{i+1}^R\}$.

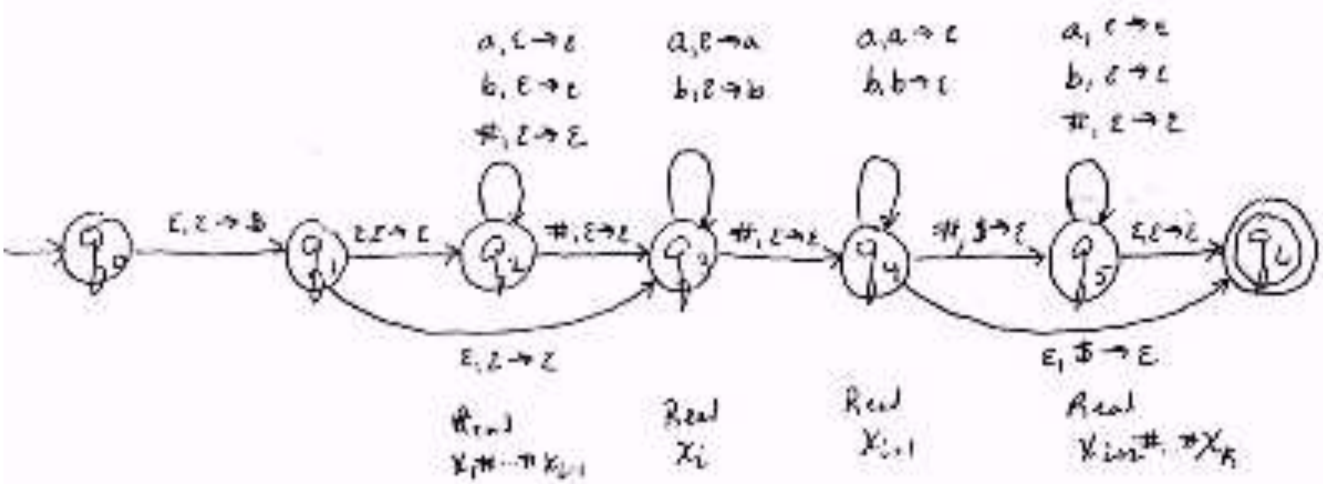
(10 pts.)

Plan:
Mark the bottom of the stack.

Nondeterministically determine if $i=1$ or $i>1$.
If $i>1$ travel through $x_1\#\ldots\#x_{i-1}$, then read #.

Read x_i, pushing a's and b's onto the stack.
Read #
Read x_{i+1}, popping a's and b's from the stack.

Nondeterministically determine if $i+1=k$ or $i+1<k$.
If $i+1=k$ pop the bottom of the stack and accept, else read #, pop the bottom of the stack, then travel through $x_{1+1}\#\ldots\#x_k$, and accept.

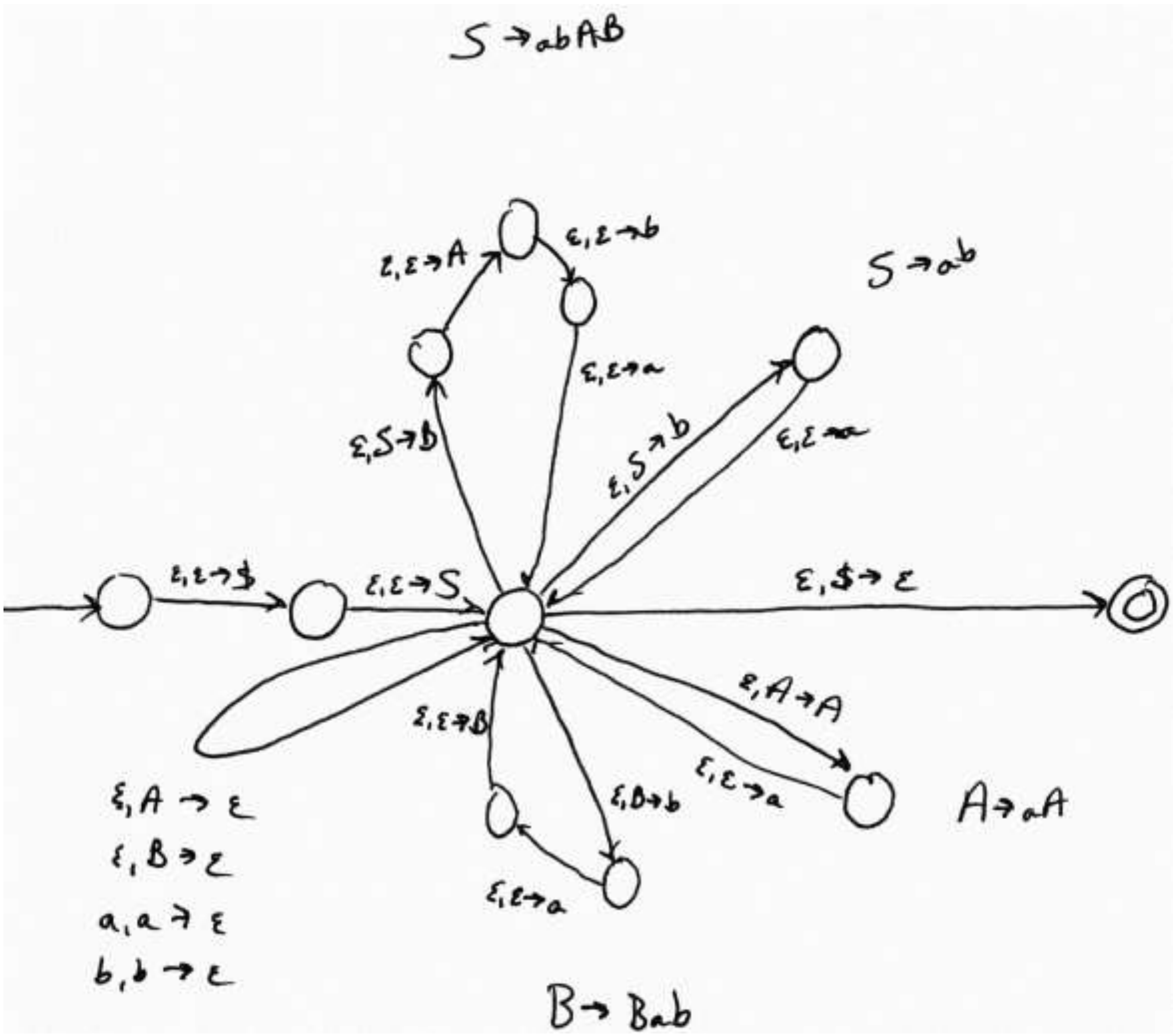


5. Using the method described in class and the text, convert the following context-free grammar into a PDA.

$$S \rightarrow abAB \mid ab$$
$$A \rightarrow a\,A \mid \varepsilon$$
$$B \rightarrow Bab \mid \varepsilon$$

(10 pts.)

Answer:



6. Convert the following grammar to Chomsky normal form.

G = ({S,T,U}, {0,#}, R , S) where R is defined by:

S→ TT | U
T → 0T | T0 | #
U → 0U00 | # (10 pts.)

Step 1. Add a new start symbol

Not needed since there is no recursion with the start symbol. However, I'll do it, just to follow the algorithm.

S'→ S
S→ TT | U
T → 0T | T0 | #
U → 0U00 | #

Step 2. Remove ε-productions

None to remove

Step 3. Remove all unit productions

There are two unit productions that need to be removed: S'→ S and S → U. For this simple grammar, the order they are removed, doesn't matter. In other grammars, it matters, but any order can be used. All orders eventually lead to a grammar in Chomsky Normal Form.

S'→ TT | 0U00 | #
S→ TT | 0U00 | # Optional, since there is no path to S
T → 0T | T0 | #
U → 0U00 | #

Step 4. Add new variables as needed to get the grammar to adhere to rule of Chomsky normal form.

S'→ TT | OV | #
S→ TT | OV | # Optional, since there is no path to S
T → OT | TO | #
U → OV | #
O → 0
V → UW
W → OO

7. Consider the language $L = \{ ww \mid w \in \{a, b\}^* \}$. Prove that L is context-free by giving a grammar for it, or use a pumping lemma to prove L is not a context-free language. (20 pts.)

Claim L is not context-free.

Proof: Suppose, by way of contradiction, that L is context-free. Then the pumping lemma must hold for L. Let p be the pumping length. Consider the string s= "$a^pb^pa^pb^p$". Clearly $s \in L$ and $|s| \geq p$. The pumping lemma guarantees that s can be divided into five pieces s=uvxyz where $|vxy| \leq p$, $|vy|>0$ and $s_i=uv^ixy^iz \in L$ for all $i \geq 0$.

Consider all possible breakdowns s=uvxyz where $|vxy| \leq p$, $|vy|>0$.

Case 1: vxy is entirely in one set of symbols, either the first set of a's, the first set of b's, the second set of a's, or the second set of b's.
Case 2: v only contains one type of symbol and y only contains the other type of symbol (there are three possibilities: between the first set of a's and the first set of b's; between the first set of b's and the second set of a's; or between the second set of b's and the second set of a's)
Case 3: either v or y contains both an 'a' and a 'b'.

Consider each case separately.

Case 1: vxy is entirely in one set of symbols, either the first set of a's, the first set of b's, the second set of a's, or the second set of b's.
Consider $s_0=uv^0xy^0z$.

If vxy is in the first set of a's, s_0 ="$a^{p-|vy|}b^pa^pb^p$" and $s_0 \notin L$.
If vxy is in the first set of b's, s_0 ="$a^pb^{p-|vy|}a^pb^p$" and $s_0 \notin L$.
If vxy is in the second set of a's, s_0 ="$a^pb^pa^{p-|vy|}b^p$" and $s_0 \notin L$.
If vxy is in the second set of b's, s_0 ="$a^pb^pa^pb^{p-|vy|}$" and $s_0 \notin L$.

Case 2: v only contains one type of symbol and y only contains the other type of symbol.
Consider $s_0=uv^0xy^0z$.

If v is in the first set of a's and y is in the first set of b's,
s_0 ="$a^{p-|v|}b^{p-|y|}a^pb^p$" and $s_0 \notin L$
If v is in the first set of b's and y is in the second set of b's,
s_0 ="$a^pb^{p-|v|}a^{p-|y|}b^p$" and $s_0 \notin L$
If v is in the second set of a's and y is in the first set of a's,
s_0 ="$a^pb^pa^{p-|v|}b^{p-|y|}$" and $s_0 \notin L$

Case 3: Either v or y contains both an 'a' and a 'b' (it is impossible for both v and y to contain both an 'a' and a 'b' because $|vxy| \leq p$).

If v or y contains both one or more ‘a’s and one or more ‘b’s from the first set of a’s and b’s, the string s_2 will have three sets of b’s, rather than two. A string with three sets of b’s cannot be of the form ww.

Similarly, if v or y contains both one or more ‘b’s from the first set of b’s, and one or more ‘a’s from the second set of a’s.

Similarly, if v or y contains both one or more ‘a’s from the second set of a’s, and one or more ‘b’s from the second set of b’s.

It has been shown that there is no way to divide s into uvxyz so the pumping lemma holds. Thus, the pumping lemma does not hold for L, and L must not have been context-free.

Extra Credit

Consider the language $L=\{a^n b^j \mid n = j^2\}$. Prove that L is context-free by giving a grammar for it or use the pumping lemma for context-free languages to prove that there is no context-free grammar for L.

L is not context free.

Proof: Suppose, by way of contradiction, that L is context-free. Then the pumping lemma must hold for L. Let p be the pumping length. Consider the string s= "$a^{p^\wedge 2} b^p$". Clearly $s \in L$ and $|s| \geq p$. The pumping lemma guarantees that s can be divided into five pieces s=uvxyz where $|vxy| \leq p$, $|vy|>0$ and $s_i = uv^i xy^i z \in L$ for all $i \geq 0$.

Consider all possible breakdowns s=uvxyz where $|vxy| \leq p$, $|vy|>0$.

Case 1: vxy is entirely in the a's.
Case 2: vxy is entirely in the b's.
Case 3: v or y contains both an 'a' and a 'b'.
Case 4: v is only in the a's and y is only in the b's.

Consider each case separately.

Case 1: vxy is entirely in the a's.
$s_2 = uv^2xy^2z = a^{p^\wedge 2+|vy|} b^p$ which is not in L.

Case 2: vxy is entirely in the b's.
$s_2 = uv^2xy^2z = a^{p^\wedge 2} b^{p^\wedge 2+|vy|}$ which is not in L.

Case 3: v or y contains both an 'a' and a 'b'.
In this case s_2 will have an 'a' appear after a 'b', which cannot be in L.

Case 4: v is only in the a's and y is only in the b's.
In this case $s_2 = a^{p^\wedge 2+|v|} b^{p^\wedge 2+|y|}$. In order for s_2 to be in L, it must be true that

$$p^2+|v| = (p^2 + |y|)^2$$

However, $(p^2 + |y|)^2 = p^2 + 2p|y| + |y|^2$
Thus, it must be true that:

$$p^2+|v| = p^2 + 2p|y| + |y|^2$$

subtracting p^2 from each side gives:

$$|v| = 2p|y| + |y|^2$$

If $|y|>0$ then $|v|>p$ and it is not true that $|vxy| \leq p$.
If $|y|=0$ then $|v|=0$ and it is not true that $|vy|>0$.

Thus, s_2 is not in L.

It has been shown that there is no way to divide s into uvxyz so the pumping lemma holds. Thus, the pumping lemma does not hold for L, and L must not have been context-free.