Name: ______________________ Date: ___________

Note: The purpose of the following questions is:

• *Enhance learning*	• *Summarized points*	• *Analyze abstract ideas*

Class11: Pushdown Automata and Context-Free Languages

In the examples of the previous class, we saw that pushdown automata exist for some of the familiar context-free languages. This is no accident. There is a general relation between context-free languages and nondeterministic pushdown accepters that is established in the next two major results. We will show that for every context-free language there is an npda that accepts it, and conversely, that the language accepted by any npda is context-free.

Pushdown Automata for Context-Free Languages

1. Prove the Theorem: For any context-free language L, there exists an npda M such that
$$L = L(M)$$
Hint: Proof in two steps:
Step1: Convert any context-free grammar G to a NPDA M with: L(G) = L(M).
Step2: Convert any NPDA M to a context-free grammar G with: L(G) = L(M).

We first show that for every context-free language there is an npda that accept it. The underlying idea is to construct an npda that can, in some way, carry out a leftmost derivation of any string in the language. To simplify the argument a little, we assume that the language is generated by a grammar in Grreibach normal form. *It is not necessary to do this.*

The pda we are about to construct will represent the derivation by keeping the variables in the right part of the sentential form on its stack, while the left part, consisting entirely of terminals, is identical with the input read. We begin by putting the start symbol on the stack. After that, to simulate the application of a production $A \rightarrow ax$, we must have the variable A on top of the stack and the terminal a as the input symbol. The variable on the stack is removed and replaced by the variable string x. What δ should be to achieve this is easy to see. Before we present the general argument, let us look at a simple example.

2. Construct npda that accepts the language generated by a grammar with the production. Also, show the *actions* of the automaton and *leftmost* derivation at each step for the string *abab*.
$$S \rightarrow aSTb$$
$$S \rightarrow b$$
$$T \rightarrow Ta$$
$$T \rightarrow \lambda$$

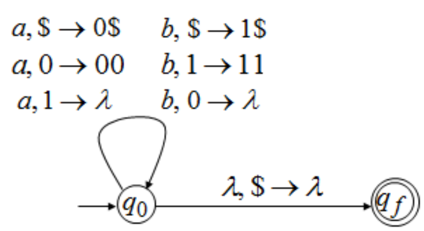
3. ***Show***: in general, for any given grammar G, how can we construct a NPDA M with

$$L(G) = L(M)$$

4. *In step2: Convert any NPDA M to a context-free grammar G with: L(G) = L(M).*
   The converse of *step 1* is also true. The construction involved readily suggests itself: Reverse the process in *step 1* so that the grammar simulates the moves of the pda. This mean that the content of the stack should be reflected in the variable part of the sentential form, while the processed input is the terminal prefix of the sentential form. Quite a few details are needed to make this work. **Explain**.
5. In step 2: **Explain** the following:
   Modify (if necessary) the NPDA so that:
   1) The stack is never empty
   2) It has a single final state and empties the stack when it accepts a string
   3) Has transitions in a special form.
6. Find a context-free grammar that generates the language accepted by the following npda M

$$L(M) = \{w : \; n_a = n_b\}$$

$\$$: initial stack symbol



   Using the resulting grammar, show the derivation of the string *abba*
7. Use the previous npda example to conclude the argument of step 2 in your proof.

Deterministic Pushdown Automata:

A deterministic pushdown accepter (DPDA) is a pushdown automaton that never has a choice in its move.

8. Give examples of allowed transitions in DPDA.
9. Give examples of *not* allowed transitions in DPDA.
10. Give example of DPDA and show the language.
11. Give example of NPDA and show the language.
12. Define deterministic context-free language.
13. From pervious and the next example we see that, in contrast to finite automata, deterministic and nondeterministic pushdown automata are *not* equivalent. There are context-free languages that are not deterministic. **Explain**
14. Prove the **Theorem:** the Language $L=\{a^n b^n\} \cup \{a^n b^{2n}\}$ is not deterministic context-free (i.e. there is no DPDA that accepts L)