Test 2 for CS 3350 Automata, Fall 2022

1-3. Let us consider a finite automaton that checks whether a dog is hungry. Let us consider an alphabet consisting of two symbols: f (for "*f*ood"), and n (for "*n*o food"). This automaton has two states:

- the start state h (for "*h*ungry") and
- the final state s (for "*s*atisfied").

Transitions are as follows:

- from the state h, f leads to s, while n lead back to h;
- from the state s, every symbol leads back to s.

This automaton accepts the word nfn.

1. Show how the general algorithm will produce a context-free grammar that generates all the words accepted by this automaton – and only words generated by this automaton.
2. On the example of the word nfn accepted by this automaton, show how the tracing of acceptance of this word by the finite automaton can be translated into a generation of this same word by your context-free grammar.
3. Show how the word nfn can be represented as $uvxyz$ according to the Pumping Lemma for context-free grammars.

4-6. Let us consider the grammar with the starting variable H and the rules $H \to nfS$, $S \to n$, $S \to f$, and $H \to \varepsilon$.

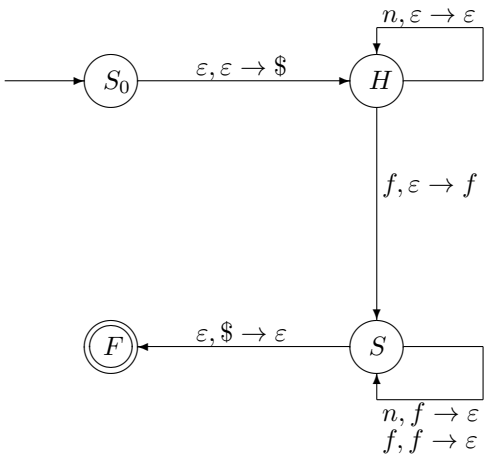
4. Use a general algorithm to construct a (non-deterministic) pushdown automaton that corresponds to this grammar.
5. Show, step by step, how the word nfn will be accepted by this automaton.
6. Transform this grammar into Chomsky normal form.

7-8. Show, step by step:

7. how the stack-based algorithm will transform the expression $a/b - c \cdot d$ into a postfix expression, and then

8. how a second stack-based algorithm will transform this postfix expression into quadruples.

9-10. Let us consider the following pushdown automaton:



This pushdown automaton accepts the word nfn. Use the general algorithm to show how this word will be generated in the corresponding context-free grammar.