# Test 2 for CS 3350 Automata, Spring 2022

1-3. Let us consider a finite automaton that checks whether the student knows the material. Let us consider an alphabet consisting of two symbols: $s$ (for "*s*tudy"), and $d$ (for "*d*o not study"). This automaton has two states:

- the start state $n$ (for "*n*o knowledge") and
- the final state $k$ (for "*k*nowledge").

Transitions are as follows:

- from the state $n$, $s$ leads to $k$, while $d$ lead back to $n$;
- from the state $k$, every symbol leads back to $k$.

This automaton accepts the word $dsd$.

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 $dsd$ 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 $dsd$ 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 $N$ and the rules $N \to dsK$, $K \to d$, $K \to s$, and $K \to \varepsilon$.

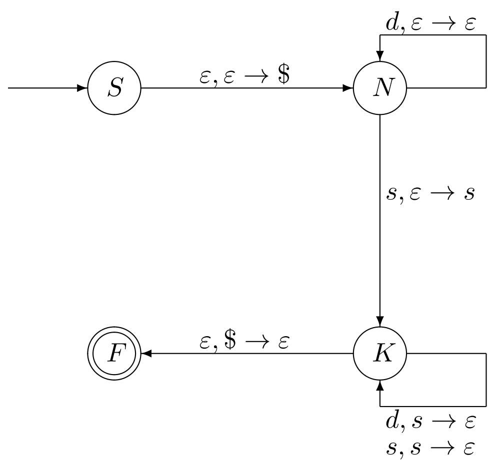
4. Use a general algorithm to construct a (non-deterministic) pushdown automaton that corresponds to this grammar.
5. Show, step by step, how the word $dsd$ will be accepted by this automaton.
6. Transform this grammar into Chomsky normal form.

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 *dsd*. Use the general algorithm to show how this word will be generated in the corresponding context-free grammar.