1. The file

/home/reseda/computational-linguistics/code/fsa-example.ss

walks you through the construction and compilation of a finite-state automaton that recognizes Sheeptalk. Copy it into the directory you use for this course, bring it up in DrRacket, read through it, and run it.

In DrRacket’s “Interactions” window, use the sheep-compiled procedure to determine which, if any, of the strings "ba!", "baaaaa!", "baa!", and "baa!!" the finite-state automaton accepts. Turn on tracing to follow the step-by-step operation of the automaton.

2. An utterance in the Cowtalk language comprises one or more “segments,” separated by spaces. A segment begins with one or more ms, followed (optionally) by a w, followed by two or more os, followed by zero, one, or two hs and an exclamation point. The shortest Cowtalk utterance is therefore "moo!", and a longer one is "mwoooh! mmmwooh! mmmmmmmooooo!".

   Draw the state diagram for an automaton that recognizes the Cowtalk language.

3. Write a Scheme expression that builds the transition function for Cowtalk.

4. In Scheme, define a finite-state automaton that recognizes Cowtalk, compile it, and test the compiled procedure.

Copyright © 2018 John David Stone

This work is licensed under the Creative Commons Attribution-ShareAlike 4.0 International License. To view a copy of this license, visit

http://creativecommons.org/licenses/by-sa/4.0/deed.en_US

or send a letter to Creative Commons, 543 Howard Street, 5th Floor, San Francisco, California, 94105, USA.