4TB3 Proposal: Exercise Correctness

Joseph Seger, Fawaz Ismail

February 2014

1 Overview

For the project, we will take the approach of making an informative and interactive webpage designed to teach the basic concepts behind finite state machines to those with little or no previous experience with the topic.

The format of the webpage will be chapter by chapter and will progress from the first chapters covering the more basic concepts on to more and more challenging topics. There will be approximately five to seven chapters.

Each chapter will contain a title of the topic to be covered, followed by an in-depth explanation. In order to provide the best possible learning experience, the explanation will be simple and easy to follow. This will also include diagrams to demonstrate each section. Upon completion of the chapter, the user will be prompted to complete an optional challenging exercise to test their comprehension and understanding. The user's answer will be checked for correctness and feedback will be provided. If the user completes two unsuccessful attempts to the exercise, they will be provided with the option to view a possible answer.

The real difficult aspect of the assignment will be creating code that checks the correctness of the reader's exercise answers. The code for each exercise will be written in Javascript and will be robust enough to check the correctness of many different problems of the same type (e.g. multiple state minimization problems all based on different finite state machines). Each type of problem will have its own unique correctness checker.

2 Chapter Topics

Topics that will be discussed in the ebook will all relate to finite state machines in some way. The following topics will be included:

- Regular Grammars - What they are and how they relate to finite state machines.
  Exercises: Design a grammar which produces a given language/describe the language a given grammar would produce.
- Finite State Machines - Construction and mechanics.
  Exercises: Translate a given finite state machine into a regular grammar/construct a finite state machine given a regular grammar.
- State minimization - Concept behind "redundant" states.
  Exercise: Minimize a given finite state machine.
- Determinism - Concept behind determinism vs nondeterminism.
  Exercise: Create an equivalent deterministic finite state machine given a nondeterministic finite state machine.