CSE 4083/5210 Formal Languages and Automata

`asgn02` Assignment #2 Chapter 2 (Regular/FA)

Due: Fri, 19 Jan 2024

Reading. Read Chapter 2: "Finite Automata" from the textbook. Omit Section 2.4. The textbooks has it all, but lecture notes can be found at the class WWW site. On Canvas LMS there are some pre-recorded lectures, but *study the textbook*.

Assignment. Do some number of the following exercises from Linz (the same numbering is used in the sixth and seventh edition):

- Section 2.1: All the problems are good, e.g., 7(a–g). Also 13–18. Proofs: 19, 21, 24, 26.
- Section 2.2: Problems 5–7, 13 (easy); 8, 9, 11a, 12.
- Section 2.3: Problems 1–5. Proofs/constructions: 13, 14, 16, 17, 18.
- Section 2.4: Problems 1–3(a–e). Proofs: 7.

We are especially interested in clear exposition and proof technique. (Some solutions sketches are in the back of the book.)

Submission. Write up the solutions. You may use pen and paper, plain text, or LaTeX. Produce a PDF document, and submit it on Canvas by the due date before the end of the day.

The due date is for the completed problem set. You should read the material in advance, and start thinking and working on the problems in advance, so that you can ask questions in class.

Collaborating is encouraged; no individual grade for the homework will be used in determined the individual course grade (that's what the tests are for). Copying just wastes everyone's time; it is quality that is important not quantity. Copying is not practicing. Of course, some individual may require much more practice than others to achieve the same level of competency on the tests.

Questions. If you have questions about how to do the problems, you are welcome to send me e-mail: `ryan@fit.edu`. Students may be called upon to share and explain their progress on the exercises during class.

Assessment. Ultimately the written proofs, your choice of exercises, and your participation in answering and *asking* questions, will influence your course grade.

Objectives. At the conclusion of the chapter, the student should be able to the following.

1. (§2.1) Understand the definition of a deterministic finite accepter (dfa) (Definition 2.1, page 39)
2. (§2.1) Demonstrate whether an input string is accepted by a dfa and describe the language accepted by a dfa (Definition 2.2, page 41)
3. (§2.1) Construct a dfa to accept a specific language (Example 2.3, page 44)
4. (§2.1) Show that a particular language is regular (Definition 2.3, page 46)
5. (§2.2) Demonstrate whether an input string is accepted by a nfa and describe the language accepted by a nfa (Definition 2.6, page 55)
6. (§2.2) Compare and contrast deterministic and nondeterministic finite automata (nfa)
7. (§2.2) Construct a nfa to accept a specific language
8. (§2.3) Transform an arbitrary nfa to an equivalent dfa (Example 2.12, page 60)
9. (§2.4∗) Transform an arbitrary dfa to an equivalent dfa with the minimum number of states