Reading. Read Chapter 11 “A Hierarchy of Formal Languages and Automata” and Chapter 12 “Limits of Algorithmic Computations.” There are several pre-recorded lectures pertaining to this assignment. They can be found following the links on the grid of notes or on the Canvas LMS.

Assignment. Do some of the following exercises.

- Section 11.1: Problems 3, 4, 5–10, 11.
- Section 11.2: Problems 1, 3, 5.
- Section 11.3: Problems 2abcde, 3ab.
  When giving a grammar be sure to include sample derivations and insightful verbal explanations of the purpose and use of the nonterminals.
- Section 11.4: Problems 1.
- Section 12.1: Problems 2, 3, 5.

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 end of the day Fri, 19 Nov 2021.

Questions. If you have questions about how to do the problems, you are welcome to send me e-mail: ryan@fit.edu. Starting on Monday students will be called upon to share and explain their answers during class time.

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

Objectives.

1. Explain the difference between recursive and recursively enumerable languages
2. [Countable sets, diagonalization technique]
3. Describe the type of productions in an unrestricted grammar
4. Identify the types of languages generated by unrestricted grammars

5. Describe the type of productions in a context sensitive grammar

6. Give a sequence of derivations to generate a string using the productions in a context sensitive grammar

7. Identify the types of languages generated by context-sensitive grammars

8. Construct a context-sensitive grammar to generate a particular language

9. Describe the structure and components of the Chomsky hierarchy; [CSL proper subset of REC by diagonalization; HU 1969, Section 8.3, page 117]

1. Define the Turing machine halting problem

2. Give examples of undecidable problems regarding recursively enumerable languages