CS 374: Algorithm and Models of Computation (Spring 2018)
Syllabus for Midterm 1

The first midterm will test material covered in lectures 1 through 8. This corresponds to material in Jeff's notes that is Appendix I (induction), Chapter 1 (strings), Chapter 2 (regular languages and expressions), Chapter 3 (DFAs) except section 3.6 (Decision Algorithms), Chapter 4 (NFAs) except sections 4.7 through 4.10, Chapter 2.5 (Context-free Languages), and Chapter 36 upto section 36.3 (Turing machines). Additional material covered in the lectures is in notes titled "more induction notes", "DFA proof notes", "NFA notes", and "Infinite fooling set Guide". Any results proved in discussion labs and homework assignments are also part of material that can be tested on the midterm.

Specific skills that may be test include (the following list may not be exhasutive)

Basic mathematics

- Comfort with set notation, especially set operations like cross product and power set. Should know how to read and understand formally described sets, and should be able to describe new sets precisely.
- Familiarity with alphabets, strings, and languages.
- Ability to critically evaluate proofs and write proofs, especially induction proofs.
- Ability to comprehend inductive definitions.

Formal models of computation (regular expressions, DFAs, NFAs, CFGs, Turing machines)

- Understand formal definitions of machines, grammars and expressions. Be able to execute machines on simple examples, and infer if strings belong to sets defined by expressions/grammars. Understand what it means for a language to be described/accepted by a computational model.
- Ability to design machines/grammar/expressions to describe/accept languages. Ability to formally describe them.

Transformations between computational models

- Familiarity with proofs transforming NFAs to DFAs, and regular expressions to NFAs. Ability to carry out these constructions on examples.
- Familiarity with the cross product construction to run multiple machines simultaneously.
- Know assymptotic bounds of the resulting automata constructed by these transformations.
- Ability to perform new transformations on automata to prove regularity or construct automata/expressions with special properties.

Closure properties

- Know standard closure properties (union, intersection, complementation, homomorphisms, inverse homomorphisms) for regular languages covered in lectures, labs and homework. Understand the proofs for these properties.
- Know how to prove new closure properties either through automata tranformations or using previously established closure properties.

Non-regularity

- Ability to distinguish regular and non-regular languages

- Ability to prove languages to be non regular using the fooling set argument. Know how to prove lower bounds on the number of DFA states using the fooling set argument as well.

Decidability and recursive enumerability

- Understand the definitions of Turing machines accepting, rejecting, halting, and not halting.

- Understand the definitions of Turing machines recognizing a language and deciding a language.

- Understand the definitions of decidability and recursive enumerability.