The first exam is in class Tuesday April 25, 2017. The assigned seats are available on Piazza. You must take the exam with the section in which you are enrolled. The exam is closed books, closed notes, except that you may bring and use one (double-sided) handwritten standard sized (3 inch by 5 inch) index card. Each student may use only their own index card; no collaboration during the exam is allowed. The exam covers all the material up to and including the lecture immediately before the exam. In particular, you should study:

- Sipser Chapters 0 and Sections 1.1 - 1.3.
  (More emphasis on Chapter 1, since Chapter 0 is a review of material you are expected to know already.)

- The main concepts we discussed include
  Sets, strings, languages, functions, graphs, Boolean logic, proofs, state machines, finite automata (DFA), computation traces, regular languages, closure (of arbitrary sets under arbitrary operations, and of the set of regular languages in particular), non-determinism, power-set construction, equivalence of automata, regular expressions.

- Recommended exercises from the textbook (question references are to the 3rd edition, International version):
  - 0.2, 0.3, 0.8, 0.13
  - 1.4b, 1.4d, 1.5a, 1.6g, 1.7a, 1.7f, 1.10a, 1.19, 1.20, 1.45a.

- All assigned homework questions, as well as suggested questions from Discussions.

  It’s a good idea to go over the feedback you received on the homework, and also look over the homework solutions posted on Piazza.

  You can expect to see some questions on the exam that are similar to homework questions.

Below are questions indicative of the general style you might expect to see on the exam. You might find working through them helpful for studying purposes. Solutions will not be distributed for these practice problems (other than the ones already available for homework). It is often much more productive to figure out for yourself how to start a problem and then ask for help if and when you get stuck.
(1) Give state diagrams for DFAs recognizing the following languages over the alphabet \( \{a, b\} \). (For full credit, use at most the given number of states and briefly justify your DFA by giving an informal description of the construction.)

(a) \( \{w \in \{a, b\}^* \mid w \text{ is any string except } aa \text{ and } aaa\} \). (This can be solved with 5 states.)

(b) \( \{w \in \{a, b\}^* \mid w \text{ contains an even number of } a\text{s, or contains exactly two } b\text{s}\} \)
(This can be solved with 8 states.)

(2) Consider the NFA with state diagram

(a) Describe precisely the language (over \( \Sigma = \{0, 1\} \)) recognized by this NFA.
(b) Draw the state diagram of a DFA that recognizes the same language.

(3) Fix an alphabet \( \Sigma \). For languages \( A \) and \( B \) (both over \( \Sigma \)), let the perfect shuffle of \( A \) and \( B \) be the language

\[ \{w \in \Sigma^* \mid w = a_1b_1 \cdots a_kb_k, \text{ where } k \geq 0, \text{ each } a_i, b_i \in \Sigma \text{ and } a_1 \cdots a_k \in A \text{ and } b_1 \cdots b_k \in B.\} \]

Prove that the class of regular languages (over \( \Sigma \)) is closed under perfect shuffle.

(4) Prove that every NFA can be converted to an equivalent one that has no incoming transitions to its initial state.

(5) List the following regular expressions in order from left to right so that regular expression \( X \) is to the left of regular expression \( Y \) whenever \( L(X) \subseteq L(Y) \).

(a) \( 0^*1^* \)
(b) \( 0001^* \)
(c) \( (0^*1^*)^* \)
(d) \( 1^*0^*1^* \)