Midterm Exam 2 Study Guide

CSE 30151 Spring 2017

2017/04/05

You will have the whole class period of 75 minutes. The exam will be open book and open (paper) notes. No computers, smartphones, or tablets will be allowed. The exam covers HW3–6. There will be six questions, worth 10 points each, for a total of 60 points (10% of your grade).

Many of the practice problems below are from the textbook. The numbers are from the 3rd US edition. If the 3rd international edition has a different number, it is indicated by "intl."

1. Prove that a language is nonregular. You can use the pumping lemma, any results proved in the book or in class, or any combination thereof. Like HW3 2b, 3b; Sipser 1.29ac, 1.46b (intl. 1.51b).

2. Write a CFG *and* a PDA that recognizes a language. For the PDA, a formal description (equations or state diagram) is required. Like HW4 1ab, 3b; Sipser 2.4ad, 2.6ac, 2.7ac.

3. Given some operation on languages, prove that CFLs are closed or not closed under this operation, or that this operation turns regular languages into CFLs. Like HW5 2ab (but not as hard), 3a; Sipser 2.38 (intl. 2.50; but not as hard).

4. Prove that a language is not context-free. You can use the pumping lemma, any results proved in the book or in class, or any combination thereof. Like HW5 1abc; Sipser 2.30bc (intl. 2.42bc).

5. Write a Turing machine that decides a language. A formal description (equations or state diagram) is required. Like HW6 1ab; Sipser 3.8a.

6. Prove that a kind of machine is equivalent to Turing machines. Only implementation-level descriptions are needed, and we'll provide a template walking you through the proof (see next page). Like HW6 2; Sipser 3.10 (intl. 3.17).

Example template for HW6 Q2 and its converse A Turing machine with a doubly infinite tape is like a TM as defined in the book, but with a tape that extends infinitely in both directions (not just to the right). Initially, the head is at the first symbol of the input string, as usual, but there are infinitely many blanks to the left. Show that TMs with doubly infinite tapes are equivalent to TMs.

(a) Given a standard TM $S = (Q, \Sigma, \Gamma, \delta, q_0, q_{\text{accept}}, q_{\text{reject}})$, construct an implementation-level description of a doubly-infinite TM D:

- If S's tape is $t_1 t_2 t_3 \cdots$, where $t_i \in \Gamma$, and $h \geq 1$ is the head position, what would the corresponding configuration of D look like?
  Solution: The tape would be $\cdots \sqcup\sqcup\sqcup \texttt{\#} t_1 t_2 t_3 \cdots$, where # is at position 0, and the head would be at position h.
- How should D simulate reading symbol a?
  Solution: Just read a.
- How should D simulate writing symbol b?
  Solution: Just write b.
- How should D simulate moving to the left?
  Solution: Move to the left. Then if the head is on #, move back to the right.
- How should D simulate moving to the right?
  Solution: Just move to the right.

(b) Given a doubly-infinite TM $D = (Q, \Sigma, \Gamma, \delta, q_0, q_{\text{accept}}, q_{\text{reject}})$, construct an implementation-level description of a standard TM S:

- If D's tape is $\cdots t_{-2} t_{-1} t_0 t_1 t_2 \cdots$, and h is the head position, what would the corresponding configuration of S look like?
  Solution: The tape would be $\texttt{\#} t_\ell \cdots t_r \sqcup\sqcup\sqcup \cdots$, where ℓ is the leftmost nonblank square in D's tape and r is the rightmost nonblank square in D's tape.
- How should S simulate reading symbol a?
  Solution: Just read a.
- How should D simulate writing symbol b?
  Solution: Just write b.
- How should S simulate moving to the left?
  Solution: Move to the left. Then if the head is on #, insert a blank symbol immediately after #, shifting all other symbols to the right, and leave the head on the new blank cell.
- How should S simulate moving to the right?
  Solution: Just move to the right.

Another Turing machine question Here's an example of a problem that asks you to write the formal description of a Turing machine.

Write a Turing machine that recognizes the language

$$L = \{u\texttt{\#}v \mid u \text{ and } v \text{ are binary numbers of equal length and } u \text{ and } v \text{ are bitwise complements}\}$$

Binary numbers should be nonempty and can be zero-padded on the left.

For example, `1010#0101` belongs to L, but the following strings do not:

`#`	numbers are empty
`1#1`	not bitwise complements
`1010#101`	numbers are not same length

Solution We leave the reject state implicit. We adopt the book's convention that $a, b \to \mathrm{R}$ means "on input symbol a or b, move right," but for clarity we'll write it as $a \mid b \to \mathrm{R}$.