Final Exam Study Guide

CSE 30151 Spring 2018

2018/05/01

The final exam will be on May 8, 10:30am–12:30pm, in our usual classroom, 129 DeBartolo. The exam will be open book and open (paper) notes. No computers, smartphones, or tablets will be allowed. The exam will be comprehensive. There will be eight questions, worth 15 points each, for a total of 120 points (20% 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."

Questions 1–5 will present you with five languages:

1. Regular: Prove that a given language is regular, by writing (your choice of) a finite automaton or regular expression. Like HW2 1, 2a; HW3 2a, 3a.
2. Context-free
   (a) Prove that a given language is *not* regular, using the pumping lemma, any results proved in the book or in class, or any combination thereof. Like HW3 1b, 2b, 3b; Sipser 1.29ac, 1.46b (intl. 1.51b).
   (b) Prove that the same language is context-free, by writing (your choice of) a CFG or a PDA. Like HW4 1ab, 3c; Sipser 2.4ad, 2.6ac, 2.7ac.
3. Deterministic polynomial time
   (a) Prove that a given language is *not* context free, using the pumping lemma, any results proved in the book or in class, or any combination thereof. Like HW5 1; Sipser 2.30bc (intl. 2.42bc).
   (b) Prove that the same language is in P by writing a formal description or implementation description (your choice) of a Turing machine that decides it and giving a brief complexity analysis. Like HW6 1ab; Sipser 3.8a.
4. NP-complete: Prove that a language is NP-complete, using a reduction from another language known to be NP-complete (like HW8 Q1–3).
5. Undecidable: Prove that a language is undecidable, using a reduction from another language known to be undecidable (like HW7 Q1a, Q3a).

Questions 6–8 could be about anything, including but not limited to:

- Convert a NFA to a DFA (like HW2 Q2c, Sipser 1.16), or a regular expression to a NFA (like Sipser 1.28), or a NFA to a regular expression (like Sipser 1.21), or a CFG to a PDA (like HW 4 2a), or a PDA to a CFG (like HW 4 2b).
- Given some operation on languages and a language class, prove that that language class is/isn't closed under that operation. For example:
  - regular languages (like HW2 Q3, Sipser 1.31 (intl. 1.36), Sipser 1.66a (intl. 1.60)).
  - context-free languages (like HW5 2, but not as hard), 3; Sipser 2.38 (intl. 2.50), but not as hard).
  - other language classes are possible as well: P, NP, decidable, or Turing-recognizable.
- Given some variation of finite automata, regular expressions, CFGs, PDAs, or TMs, or some kind of new machine or model of computation, show that it is/isn't equivalent to finite automata, regular expressions, CFGs, PDAs, or TM. Like HW3 1ab; HW6 2, 3; HW 7 1b.
- Church-Turing Thesis: what does it say, and why should or shouldn't we believe it?
- Algorithms for finite automata and regular expressions (CP1–2)

Topics *not* covered

- Deterministic context-free languages (Sipser, 2.4)
- Algorithms for context-free grammars (Sipser, pages 108–110, 198–200)
- Reductions via computation histories (Sipser, pages 221–233)
- Advanced topics in computability theory (Sipser, Chapter 6)
- The proof of the Cook-Levin Theorem (Sipser, pages 304–310)
- Sipser, Chapters 8 and beyond