Computation Theory (COMP 170), Fall 2020
Final

- Answer each problem below to the best of your ability.
- Submit all parts with 72 hours of downloading the exam, and before 10:30am on Tuesday, December 22nd.
- You may use your notes, the textbook, and the Resources section of the course piazza page. **No other resources may be used. Your work must be entirely your own.**
- Your submission must follow the formatting guidelines given at the end of this document.

---

[1] (*8 pts.*) **Shorties**

(a) The 4-Coloring problem ($\mathsf{4COL}$) is as follows: Given a graph $G = (V, E)$, can all the nodes of G be assigned one of four colors such that no two adjacent nodes are assigned the same color? Give a construction showing that

$$\mathsf{3COL} \leq_p \mathsf{4COL}$$

No proof is needed, nor do you need to argue that $\mathsf{4COL}$ is in NP.

(b) Consider the following attempt at showing that $HALT_{TM} \leq_m \overline{HALT_{TM}}$:

Given a TM M and string x, construct a new TM $\hat{M}$ as follows.
$\hat{M} =$ " on input y:
1. Run M on y.
2. If M halts on y, $\hat{M}$ enters a trivial loop; otherwise, $\hat{M}$ halts and accepts y."

What is wrong with this solution?

(c) Suppose I prove that A is in the class NP, and I find a polynomial-time algorithm for A. Can I conclude that P = NP? Explain.

(d) Give a regular expression for all non-empty strings over $\{a, b\}$ that start and end with the same character. No justification needed.

[2] (*8 pts.*) **Mystery Grammar**

Consider the grammar G with production rules

$$S \to aS \mid Sb \mid bSa \mid \epsilon.$$

Specify the language generated by this grammar, and prove your answer.

[3] *(8 pts.)* **Balanced Concat**

Given languages A and B, define $A \heartsuit B = \{xy \mid x \in A, y \in B, |x| = |y|\}$.

(a) Specify two languages A and B such that both A and B are regular but $A \heartsuit B$ is *not* regular.

(b) Suppose that A and B are regular sets accepted by DFAs $M_A = (Q_A, \Sigma, \delta_A, s_A, F_A)$ and $M_B = (Q_B, \Sigma, \delta_B, s_B, F_B)$ respectively.

Show that $A \heartsuit B$ is context free by constructing an NPDA M that accepts $A \heartsuit B$. You don't need to prove correctness, but your construction must be precise, and include an English explanation of what M does.

[4] (*8 pts.*) **Self-Acceptance**

Define the language

$$C = \{\langle M\rangle \mid M \text{ is a Turing machine that accepts } \langle M\rangle\}.$$

That is, C is the language of all Turing machines that accept their own description. Note that these machines are allowed to accept other strings as well, but must accept their own description.

(a) Does Rice's Theorem apply here? Why or why not?

(b) Prove that C is Turing-recognizable, but not decidable.

[5] (*8 pts.*) **Box Packing**

Suppose you've got a bunch of 2D boxes that you plan on delivering to all the good little 2D children. Your delivery vehicle of choice happens to also be rectangular (and 2D). To maximize efficiency, you'd like to completely fill your delivery rectangle, if possible. Can this be done?

More precisely, in the Box Packing problem, you've got a list of n boxes, and box i has width w_i and height h_i. You've also got your sled/van/dirigible, which has width W and height H. Assume all of these values are positive integers. The question: can you find a subset of the boxes such that those boxes perfectly fill your vehicle? That is, the boxes shouldn't overlap, and there should be no remaining empty space. For simplicity, we'll assume boxes can't be rotated (they all have those "this side up" signs).

Prove the Box Packing is NP-Complete.

Hint: A special case of this problem is equivalent to one of the problems we've seen is NP-Complete. I.e. your reduction doesn't need complicated gadgets.

Format requirements: work for COMP 170 should correspond to the following guidelines:

- Work must be in type-written format, with any diagrams rendered using software to produce professional-looking results. No hand-written or hand-drawn work will be graded.
- Work must be submitted in PDF format to Gradescope.
- Each answer should start on a new page of the document. When possible, try to limit answers to a single page each.

You can find links to information about using LaTeX to produce type-written mathematical work,[1] and to a handy web-based tool for drawing finite-state diagrams, on the Piazza class site:

https://piazza.com/tufts/fall2020/comp170/resources

[1] LaTeX was used to produce this document.