CPSC 421: Introduction to Theory of Computing
Assignment #6, due Tuesday November 1st by 11:59pm, via Gradescope

[8] 1. Recall that $\mathbb{Z} = \{\ldots, -2, -1, 0, 1, 2, \ldots\}$ is the set of integers. Let $\mathcal{C} = \{m, a, c, o, s\}$. Let

$$\mathcal{F} = \{ \text{ function } f \,:\, f : \mathbb{Z} \to \mathcal{C} \ \}$$

be the set of all functions whose domain is $\mathbb{Z}$ and whose co-domain is $\mathcal{C}$. Prove that $\mathcal{F}$ is uncountable by finding an injection from an uncountable set to $\mathcal{F}$.

[7] 2. Define the language

$$B_{TM} \;=\; \{ \langle M, w \rangle \,:\, M \text{ is a Turing machine that rejects input } w \,\}\,.$$

Describe a reduction $B_{TM} \leq_T HALT_{TM}$. Argue briefly that this reduction is correct.

[10] 3. Let C be any language $C \neq \Sigma^*$. Let

$$B(C) \;=\; \{ \langle N \rangle \,:\, N \text{ is a TM and } L(N) \not\subseteq C \,\}\,.$$

Show that, for every such C, the language $B(C)$ is undecidable.

Hint: Let z be any string in $\Sigma^* \setminus C$. Your argument should use z in an important way.

[10] 4. [5] a. In class we claimed that, if a polynomial-time algorithm is discovered for some problem, it is usually possible to discover a reasonably efficient algorithm, say one running in $O(n^5)$ time.

We could try to formalize this by saying that $P \subseteq TIME(n^5)$. Is this a true statement? Explain why or why not. (You may refer to theorems from class or the textbook.)

[5] b. Another standard complexity class is $E = \bigcup_{c>0} TIME(2^{cn})$. Is it true that $E = EXP$? Explain why or why not. (You may refer to theorems from class or the textbook.)

[20] 5. Let us consider a decision problem about a generalized form of Sudoku. (The case $n = 3$ corresponds to ordinary Sudoku.) A problem instance consists of an integer $n \geq 3$ and a two-dimensional grid of cells, with n^2 rows and n^2 columns. In the initial problem instance, each cell is either blank or contains a number in $\{1, \ldots, n^2\}$.

The goal is to place a number into every blank cell such that:

(1) Each column contain every number in $\{1, \ldots, n^2\}$ exactly once.

(2) Each row contain every number in $\{1, \ldots, n^2\}$ exactly once.

(3) For every $i, j \in \{0, \ldots, n-1\}$, the square at the intersection of rows $\{ni+1, \ldots, n(i+1)\}$ and columns $\{nj+1, \ldots, n(j+1)\}$ contains every number in $\{1, \ldots, n^2\}$ exactly once.

[6] a. The decision problem $SUDOKU$ is: given an initial problem instance (in which each cell could be blank or contain a number), decide whether the blanks can be filled in such that conditions (1)-(3) are satisfied. Show that $SUDOKU$ is in NP.

[9] b. Suppose that someone proves that $SUDOKU$ (the decision problem) is in P. Give a polynomial-time algorithm with the following behavior: given an initial problem instance (in which each cell could be blank or contain a number), output either:

- A *solution*, i.e., a value for each cell such that conditions (1)-(3) are satisfied, or

- "Reject" if there is no way to satisfy conditions (1)-(3).

[5] c. Suppose that the initial problem instance has half of its entries blank, and that conditions (1)-(3) can be satisfied. Suppose that a *solution* is written down in binary. How many bits does it take to write down this solution? You should explain your answer, and you can use Big-O notation.

[2] 6. **OPTIONAL BONUS QUESTION**: Fix any finite alphabet Σ. Let

$$\mathcal{A} \; = \; \{\, A \subseteq \Sigma^* \; : \; A \text{ is regular and } A \text{ has infinite cardinality} \,\}\,.$$

Prove that there is a language $C \subseteq \Sigma^*$ such that $C \cap A$ is not recognizable for all $A \in \mathcal{A}$.