Homework 5. Due November 29, 9:59PM.

CS181: Fall 2021

GUIDELINES:

- Upload your assignments to Gradescope by 9:59 PM.
- Follow the instructions mentioned on the course webpage for uploading to Gradescope very carefully (including starting each problem on a new page and matching the pages with the assignments); this makes it easy and smooth for everyone. As the guidelines are simple enough, bad uploads will not be graded.
- You may use results proved in class without proofs as long as you state them clearly.
- Most importantly, make sure you adhere to the policies for academic honesty set out on the course webpage. The policies will be enforced strictly. Homework is a stepping stone for exams; keep in mind that reasonable partial credit will be awarded and trying the problems will help you a lot for the exams.
- Note that we have a **modified grading scheme for this assignment**: A sincere attempt will get you 100% of the credit and a reasonable attempt will get you 50% for each problem. Nevertheless, please attempt the problems honestly and write down the solutions the best way you can - this is really the most helpful way to flex your neurons in preparation for the exam.
- All problem numbers correspond to our text 'Introduction to Theory of Computation' by Boaz Barak. So, exercise a.b refers to Chapter a, exercise b.

1. Exercise 9.9. Please replace NAND-RAM program with a Turing machine for the problem. That is, consider the case where $F : \{0,1\}^* \to \{0,1\}$ takes two Turing machines P, M as input and $F(P, M) = 1$ if and only if there is some input x such that P halts on x but M does not halt on x. Prove that F is uncomputable. [1 point]

2. Consider the function $EMPTY : \{0,1\}^* \to \{0,1\}$ that takes a DFA as input and outputs 1 if the language of the DFA is empty. That is, $EMPTY(D) = 1$ if D describes a DFA (under some encoidng - the representation is not important) that does not accept any string. Define $EQUIVALENT : \{0,1\}^* \to \{0,1\}$ as the function that takes two DFAs D, D' and checks their equivalence: that is $EQUIVALENT(D, D') = 1$ if $D(x) = D'(x)$, $\forall x$. Give a reduction from EQUIVALENT to EMPTY. [1 point]

   [You can use high-level programming languages or pseudocode to describe your reduction. By reduction, your goal is to give an algorithm R that takes an input for EQUIVALENT and outputs an input for EMPTY such that the condition for reduction holds: $R((D, D'))$ is a DFA such that EQUIVALENT$(D, D') =$ EMPTY$(R(D, D'))$. As a hint, use the closure properties of DFAs to show that there is a DFA D'' such that D'' is empty if and only if D, D' are equivalent. Then, you can argue that the DFA D" can be produced by an algorithm; you only have to provide high-level explanation for the latter.]

3. Design a context-free grammar for the following launguage: $L = \{0^m 1 0^n 1 0^{|m-n|} : m, n \geq 0\}$. Here $|m - n|$ denotes the absolute value of $m - n$.

4. Design a context-free grammar for the following language:

$$L = \{x \in \{0,1\}^* : \text{ x has an equal number of 1's and 0's}\}.$$

You can assume L has the empty string. Write a few sentences to explain your reasoning.[1 point]

[Hint: Think of cases like a) the first and last symbols of x are different; b) the first and last symbols of x are same. In case (a), you can do reduce back to the same question. In case (b), what should you do?]

Additional problems. Do not turn these in.

1. Exercise 9.13. Replace NAND-TM with just plain TM in the entire problem. [1 point]

   [Hint: Part (2) is a bit advanced but a fascinating problem - you can read up on *busy beaver function*. For part (2), try to come up with a program whose description length is at most n but that takes $\omega(TOWER(n))$ steps to stop. I also highly recommend reading the two references in the problem.]

2. Show that there is a simple constant size TM (or program in your favorite language) $Fermat$ such that the program $Fermat$ terminates if and only if the Fermat's last theorem is true (which we know it is ... but don't assume that for this problem - just show equivalence).

3. Exercise 10.1 (4), (6).

4. Design a context-free grammar for the following launguages:

   (a) $L = \{x : \text{ 5'th bit from end is a 1}\}$.

   (b) $L = \{x : x \text{ has at least three 1's}\}$.

   (c) $L = \{0^m 1^n : m \neq n\}$.

   (d) $L = \{x : x \text{ is not of the form } 0^n 1^n\}$.