CSE 311: Foundations of Computing (Spring, 2015)

Homework 8 Out: Fri, 29-May. Due: Friday, 5-Jun, **before class (1 pm)** on **Gradescope**

Additional directions: You should write down carefully argued solutions to the following problems. Your first goal is to be complete and correct. A secondary goal is to keep your answers simple and succinct. The idea is that your solution should be easy to understand for someone who has just seen the problem for the first time. (Re-read your answers with this standard in mind!) You may use any results proved in lecture (without proof). Anything else must be argued rigorously. Make sure you indicate the specific steps of your proofs by induction.

1. Relations warmup [12 points]

Determine whether the relation R on the set of all twitter users is reflexive, symmetric, antisymmetric, and/or transitive:

(a) Everyone who has retweeted user A has also retweeted user B.
(b) There are no common followers of users A and B.
(c) There is a user that users A and B both follow.
(d) There is at least one follower of user A that is also a follower of user B.

2. Relations yoga [15 points]

Suppose that R and R' are reflexive relations on a set $\mathcal{A}$. Prove or disprove each of the following statements:

(a) $R \cup R'$ is reflexive

(b) $R \cap R'$ is reflexive

(c) $R \circ R'$ is reflexive

3. Symmetry and Power [20 points]

Let R be a symmetric relation on a set $\mathcal{A}$. Use induction to show that R^n is symmetric for all integers $n \geq 1$.

4. Needs more fiber [20 points]

Let $\Sigma = \{0,1,2,3,4,5,6,7,8,9,*,=\}$.

(a) Let L be the language of all strings of the form "$x * y = z$" where $x, y, z \in \{0,1,2,3,4,5,6,7,8,9\}^*$ and $x \cdot y = z$ when the strings x, y, z are interpreted as decimal numbers. Prove that L is irregular.

(b) Let K be the language of all strings x such that the numbers of $0'$s occurring in x is prime. Prove that K is irregular.

To do part (b), you can use the pumping lemma for regular languages. It is possible without the pumping lemma, but substantially trickier. The pumping lemma is basically an advanced version of the argument we did in class—it gives you more structure on the set of strings that must arrive in the same state.

The pumping lemma can be stated as follows: Suppose that L is a regular language. Then there exists a number $p \geq 0$ (called the "pumping length") such that for every string $s \in L$, there is a decomposition $s = xyz$ satisfying the following properties

i) $|y| > 0$
ii) $|xy| \leq p$
iii) $xy^iz \in L$ for all $i \geq 0$

Basically, there is always a way to decompose a string $s \in L$ into three parts xyz such that the middle can be repeated an arbitrary number of times and the string remains in the language.

Now you should try to prove that the language $\{0^p : p \text{ is prime}\}$ is not regular, using the preceding lemma (and arguing by contradiction).

[Hint: Start with the string 0^k where k is first prime number bigger than the pumping length p, and use the pumping lemma to say that some string 0^m must be in L where m is composite.]

5. Extra credit: The pumping lemma for context free languages

Previously:

Evil Professor ImpossibleHomeworkProblem is standing at the DMV waiting to get a new license. Because people who work at the DMV are so nice and accommodating, they are just about to let him change the name on his license when…

You learn there is a "pumping lemma" for context-free languages, but it's a little complicated. If the language L has a context-free grammar, then there is some integer $p \geq 0$ such that every string $s \in L$ with $\text{len}(s) \geq p$ can be written in the form $s = uvwxy$ where u, v, w, x, y are substrings satisfying:

1. $\text{len}(vwx) \leq p$
2. $\text{len}(vx) \geq 1$
3. $uv^n wx^n y \in L$ for all $n \geq 0$

Use this lemma to prove that the language $\{1^{m+n}0^m1^n0^{m+n} : m, n \geq 0\}$ cannot be written as a context-free grammar, thereby demonstrating to everyone at the DMV that Evil Professor ImpossibleHomeworkProblem should not be allowed to change his name to "Carl."