Student ID: ______________________________

Collaborators: ______________________________

CS181 Winter 2017 – Problem Set 1

Due Thursday, January 26, 11:59 pm

- Please write your student ID **and the names of anyone you collaborated with** in the spaces provided and attach this sheet to the front of your solutions. **Please do not include your name anywhere since the homework will be blind graded.**
- An extra credit of **5%** will be granted to solutions written using LaTeX. Here is one place where you can create latex documents for free: `https://www.sharelatex.com/`. The link also has tutorials to get you started. There are several other editors you can use.
- If you are writing solutions by hand, please write your answers in a neat and readable hand-writing.
- Always explain your answers. When a proof is requested, you should provide a rigorous proof.
- 20% of the points will be given if your answer is "I don't know". However, if instead of writing "I don't know" you write things that do not make any sense, no points will be given.
- The homework is expected to take anywhere between 10 to 16 hours. You are advised to start early.
- Submit your homework in drop box A1 in BH 2432 or online on the course webpage on CCLE. You can also hand it over at the end of any class before the deadline.

> Note: *All questions in the problem sets are challenging; you should not expect to know how to answer any question before trying to come up with innovative ideas and insights to tackle the question. If you want to do some practice problems before trying the questions on the problem set, we suggest trying Exercise problems 1.4, 1.5, 1.9, 1.10, and 1.11 from the book. Do not turn in solutions to problems from the book.*
>
> The machines that we called "Finite State Machines" in class are also called "Deterministic Finite Automata (DFA)" and the machines we called "Magical Finite State Machines" in class are also called "Non-Deterministic Finite Automata (NFA)".

Hint on all construction problems: If you want to prove that L is regular, it suffices to give an NFA for it. On the other hand, if you are told to assume that L' is regular, this means that there must exist a DFA recognizing L'.

1. **(15 points)** A *Tierce NFA* is a NFA that accepts a word w if there exist computation paths for w such that more than one-third of the ending states are accepting. In this question, we will study the equivalence of tierce NFAs and DFAs.

   (a) **(5 points)** Show that for every DFA M there exists a tierce NFA N that accepts the same language.

   (b) **(10 points)** Show that for every tierce NFA N there exists a DFA M that accepts the same language.

   For each part, provide a **complete and rigorous** construction of the tierce NFA or DFA as appropriate and provide an explanation as to why your construction works.

2. **(15 points).** Let L be any language and let L_R be the set of reverse strings, i.e.

$$L_R = \{x \mid \exists y \in L \text{ such that } |y| = |x| \text{ and } x_1x_2 \dots x_n = y_ny_{n-1} \dots y_1\}.$$

   Show that, if L is regular, so is L_R.

3. **(20 points)** Let L be any language and let L_{alt} be the set of strings in L with every other character removed, i.e.

$$L_{alt} = \{x \mid \exists y \in L \text{ such that } x_1x_2x_3 \dots = y_1y_3y_5 \dots\}.$$

   Show that if L is a regular language then L_{alt} is regular.

4. **(30 points)** Let L be any language and let $L_{\frac{1}{2}-}$ be the set of all the first halves of strings in L, i.e.

$$L_{\frac{1}{2}-} = \{x \mid \exists y \in \Sigma^* \text{ such that } |x| = |y| \text{ and } xy \in L\}.$$

   Show that if L is a regular language then $L_{\frac{1}{2}-}$ is regular.
   Hint: Think about the way we implemented two machines in "parallel" by using the Cartesian product. That idea may be useful for this problem.