Homework 3: Regular expressions and non-regular languages

Theory of Computing (CSE 30151), Spring 2024

Due: **Thursday** 2024-02-15 5pm

Instructions

- Create a PDF file (or files) containing your solutions. You can write your solutions by hand, but please scan them into a PDF.
- Please name your PDF file(s) as follows to ensure that the graders give you credit for all of your work:
  - If you're making a complete submission, name it `netid-hw3.pdf`, where *netid* is replaced with your NetID.
  - If you're submitting some problems now and want to submit other problems later, name it `netid-hw3-part123.pdf`, where `123` is replaced with the problem number(s) you are submitting at this time.
- Submit your PDF file(s) in Canvas.

Problems (10 points each)

1. **Regular expressions vs. Unix regular expressions.** Regular expressions and Unix regular expressions have some superficial differences, but also some deeper ones that affect the class of languages recognized.

   (a) Unix regular expressions have *quantifiers*: if α is a regular expression, $\alpha^{\{m,n\}}$ is a regular expression that matches at least m and no more than n strings that match α. More formally, it matches all strings $w^{(1)} \cdots w^{(l)}$ where $m \leq l \leq n$, and for all i such that $1 \leq i \leq l$, $w^{(i)}$ matches α. Prove that for any regular expression with quantifiers, there is an equivalent regular expression without quantifiers.

   (b) Unix regular expressions have *backreferences*: for an explanation, please see `http://www.regular-expressions.info/backref.html`. Give an example of a Unix regular expression that uses backreferences to describe a nonregular language, and prove that this language is not regular. We want you to get practice writing a non-regularity proof, so although you may use Examples 1.73–77, do not simply cite one of them; please write out a full proof.

2. **Binary addition.** This problem is about two ways of representating addition of binary natural numbers. We consider 0 to be a natural number. We allow binary representations of natural numbers to have leading 0s, and we consider ε to be a binary representation of 0. When adding numbers, we do not allow overflow, so, for example, $1111 + 0001 = 0000$ is false.

   (a) [Problem 1.32] Let

$$\Sigma_3 = \left\{ \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \right\},$$

   that is, an alphabet of eight symbols, each of which is a column of three bits. Thus, a string over Σ_3 gives three rows of bits. Show that the following is regular:

$$B = \{w \in \Sigma_3^* \mid \text{the bottom row of } w \text{ is the sum of the top two rows}\}.$$

   Hint: Since it's easier to think about addition from right to left, design an automaton for B^R first, then convert it into an automaton for B.

   (b) [Problem 1.53] Let $\Sigma = \{\mathtt{0}, \mathtt{1}, +, =\}$, and prove that the following is not regular:

$$ADD = \{x \mathtt{=} y \mathtt{+} z \mid x, y, z \in \{\mathtt{0}, \mathtt{1}\}^* \text{ and } x = y + z \text{ is true}\}.$$

3. **Similar but different** [Problem 1.49].

   (a) Let $B = \{1^k w \mid w \in \{0, 1\}^* \text{ and } w \text{ contains at least } k \text{ } \mathtt{1}\text{s, for } k \geq 1\}$. Show that B is a regular language. Hint: Try out some strings to see what does and doesn't belong to B, in order to find another simpler way of thinking about B.

   (b) Let $C = \{1^k w \mid w \in \{0, 1\}^* \text{ and } w \text{ contains at most } k \text{ } \mathtt{1}\text{s, for } k \geq 1\}$. Prove that C is not a regular language.