HOMEWORK 6
CS 205 — Fall 2016
due SUNDAY, NOVEMBER 13th, 5am.

Please remember the rules of precedence for regular expression interpretation: alternation (|), concatenation, and star behave the same way as plus, times and exponentiation in arithmetic[1] Therefore 8 (9 | Λ) 8*9 is actually (((8 (9 | Λ)) 8*)9), when parenthesized from left to right.

1. For each of the following strings, explain precisely why (why not) they belong to the language denoted by the regular expression 8 (9 | Λ) 8*9:
   - 8989
   - 889
   - 989
   - 89

   To help in the explanations call the parts of the regular expression by name as follows: 8 is A, (9 | Λ) is B, 8* is C, and 9 is D.

2. In each of the cases below, write the *shortest/clearest* regular expression over the alphabet $\Sigma = \{b, c\}$, whose language contains exactly the sentences described. Explain why your answer is correct (i.e., why it has all the required strings, and no extra ones.) [2]
   (a) sentences with an odd number of c's and no b's
   (b) sentences with an even number of b's
   (c) sentences with every b followed by exactly two c's (so, among others, "bcb" and "bccc" cannot occur as substrings of any sentence in the language).

3. Dealing with $\emptyset$. (Recall that $\mathcal{L}(\emptyset)$ is the empty set $\{\}$.)
   - What is the shortest regular expression over the alphabet $\Sigma = \{5\}$, equivalent to $\emptyset \mid (5(\emptyset)^*)$ ["Equivalent to" in this assignment means "has the same language as".]. Give a short explanation of your answer.
   - (Challenging) Define a function *elimE* which takes as argument an arbitrary regular expression, and returns an equivalent one that has at most one occurrence of $\emptyset$. In other words $\mathcal{L}(R) = \mathcal{L}(elimeE(R))$ but $elim(R)$ has at most one occurrence of the symbol $\emptyset$. (Hint: Recall that REs are defined by structural induction. So of course, *elimE* is defined by structural induction. In addition, think about what happens if $\emptyset$ is used as the RE in various places in the recursive definition.) You will find it useful to use "if - then - else" *expressions* (e.g., *if (3 > 2) then 7 else 5* has value 7) And you can assume that you have available an equality sign = for regular *expressions*, which is just structural identity; so 0 | 1 = 0 | 1 is true but 0 | 1 = 1 | 0 is false, even though the *languages* of 0 | 1 and 1 | 0 are the same.
   - (Less challenging) In case you find the above too hard, for partial credit try this much simpler problem: define a function *erase*8 that takes a regular expression as an argument and returns another one that has the property that
     $\mathcal{L}(erase8(R))$ = { w | there is string z in $\mathcal{L}(R)$ such that w is z with all 8's replaced by the null string, i.e., 8's are erased in z}
     For example *erase*8(898) could be 9, or Λ9Λ, among others.

4. Given arbitrary regular expression R, prove formally in detail, or disprove with the smallest counter-example, each of the following 3 hypotheses[3]. In formal proofs, please use the definition of star which says that $w \in S^*$ if and only if $w \in S^j$ for some integer j, $j \geq 0$, with $S^0 = \Lambda$.

[1] Remember that $xy + z^3$ in arithmetic is interpreted as $((xy) + (z^2))$, and $x + yz^3$ is interpreted as $(x + (y(z^2)))$.
[2] As explained in class, we will not be able to grade complicated regular expressions.
[3] In the equations below, we should have written more carefully $\mathcal{L}(R^*) = \mathcal{L}(R^*R^*)$, but we'll take regular expressions to stand for the set of strings that is their language, so you can write "$w \in R$" instead of "$w \in \mathcal{L}(R)$.

(a) $R^*S^* \;==\; (RS)^*$

(b) $R^* \;==\; R^*R^*$

(c) $(RS^*)^* \;==\; \Lambda \mid R(R \mid S)^*$ (Challenging)

(Hint: Try out the above with R and S being just single letters 8 and 9. But your proof must work in the general case, so it should contain no digits! And remember that the equality proofs will have to have two parts: (a) pick arbitrary string w in the language of the LHS regexp, $\mathcal{L}$(LHS), and show it belongs to $\mathcal{L}$(RHS); (b) pick arbitrary string v in $\mathcal{L}$(RHS) and show it belongs to $\mathcal{L}$(LHS). Check out the third containment proof in the lecture notes.