Here’s how these problem-solving sessions work.

Start by working on the problem assigned to your group. If you find a solution, let me know and sketch it for me. If it’s correct, you then present it. Then write up your solution within one week, in a format I can post for the rest of the class, and send it to me. (A scanned handwritten version is fine, provided it is very neatly written.) If you do all that, you fulfill your 10% course mark allocated to these sessions.

If you quickly succeed in solving a problem (some of them are very easy!), try one of the extra problems at the end.

---

**New problems**

37. Consider the operation $\text{del}_a(w)$ that deletes all letters $a$ from the word $w$, and extend the operation to languages in the obvious way.

Give an example of a class of regular languages $L$, accepted by DFA’s with $O(n)$ states, such that the smallest DFA accepting $\text{del}_a(L)$ needs at least $2^{\Omega(n)}$ states.

39. Find the equivalence classes for the Myhill-Nerode equivalence relation for 

$$ L = \{ w \in \{0,1\}^* : |w|_0 = |w|_1 \}. $$

41. Find the equivalence classes for the Myhill-Nerode equivalence relation for 

$$ L = \{ a^{2^n} : n \geq 0 \}. $$
11. When is the concatenation of two antipalindromes an antipalindrome? Give necessary and sufficient conditions. Possible strategy: do some experiments.

13. Let $x, y$ be length-$n$ words over a $k$-letter alphabet. How many solutions are there to the equation $x \$ y = xy$?

23. Show that if $M$ is an $n$-state DFA, and accepts at least one string that is a palindrome, then it accepts a palindrome of length $< 2n^2$. Hint: using $M$ design an NFA-$\epsilon$ to accept the first halves of the palindromes in $L(M)$.

29. Let $L_1$ and $L_2$ be languages, and define the “join” operation as follows:

$$L_1 \Join L_2 = \{xyz : \text{there exists } y \neq \epsilon \text{ such that } xy \in L_1 \text{ and } yz \in L_2\}.$$  

This is similar to a “join” in database theory. Using morphisms and/or inverse morphisms show that if $L_1$ and $L_2$ are regular, so is $L_1 \Join L_2$.

33. Suppose $x \in \Sigma_k^* = \{0, 1, \ldots, k - 1\}^*$ and $y \in \Delta^*$ for some alphabet $\Delta$. If $|x| = |y|$, and $x = a_1 \cdots a_n$, $y = b_1 \cdots b_n$, we define $\text{rep}(x, y)$ to be $b_1^{a_1}b_2^{a_2} \cdots b_n^{a_n}$. Thus, for example, $\text{rep}(234, abc) = \text{aabbccc}$. Extend this definition to languages, as follows: if $L_1 \subseteq \Sigma_k^*$ and $L_2 \subseteq \Delta^*$ then $\text{rep}(L_1, L_2) = \bigcup_{|x| = |y|} \{\text{rep}(x, y)\}$. Show that if $L_1$ and $L_2$ are both regular, then so is $\text{rep}(L_1, L_2)$.