Homework 4
Com S 331, Spring 2017
Due date: **Wednesday, February 15, 2017**
Please submit the homework via BlackBoard **before the class that day**.

Note: All submissions should be **typed** and in .pdf or .doc(x) format. However, state diagrams can be drawn with hand and presented in the final manuscript as images. We recommend to use Latex for typing homeworks. You **do not** need to formally prove the correctness of your constructions unless a question specifically asks to do so.

Total points available: 100

0. Read pages 77–82 up to the end of Chapter 1 in the class-book (Sipser, 3[rd] edition).

1. **(25 points)** Let $G = (Q, \Sigma, \delta, q_{\text{start}}, q_{\text{accept}})$ be a GNFA with k states, where $k \geq 3$. The procedure *Transform(G)* presented in class is listed below (see also procedure *Convert* in the proof of Lemma 1.60 in the class-book).

   For some chosen state $q_{rip} \in (Q - \{q_{\text{start}}, q_{\text{accept}}\})$ let
   $G' = (Q', \Sigma, \delta', q_{\text{start}}, q_{\text{accept}})$ be a GNFA with $k-1$ states, such that

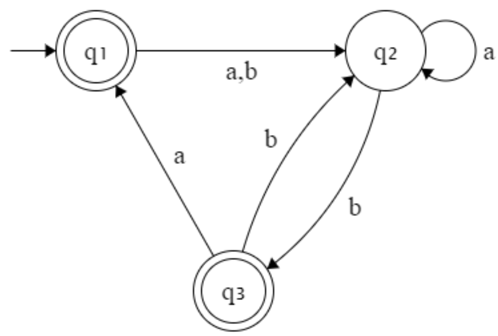
   - $Q' = Q - \{q_{rip}\}$;
   - For all $q_i \in Q' - \{q_{\text{accept}}\}$ and $q_j \in Q' - \{q_{\text{start}}\}$,

     $$\delta'(q_i, q_j) = (R_1)(R_2)^*(R_3) \cup (R_4),$$

     where $R_1 = \delta(q_i, q_{rip})$, $R_2 = \delta(q_{rip}, q_{rip})$, $R_3 = \delta(q_{rip}, q_j)$, and $R_4 = \delta(q_i, q_j)$.

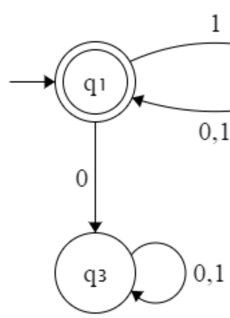
   Prove that the language of G' is equivalent to the language of G. That is, for every string $w \in L(G)$ prove that it is recognized by G' and vice versa. Please note that a **rigorous** proof is required, substantially detailing on the proof given in the class-book (as part of the inductive proof of Lemma 1.60).

2. **(25 points)** Convert the DFA (over $\Sigma = \{a, b\}$) presented on the figure below to a corresponding regular expression (see Lemma 1.60 in the class-book). Show the process of converting the DFA to a two-state GNFA step-by-step (i.e., show the initial GNFA with 5 states, then an equivalent GNFA with 4 states, and so on up to 2 states).



Note that the resulting regular expression might be quite long.

3. **(20 points)** Use induction over the size of strings in $\{0,1\}^*$ to prove that the following DFA



recognizes the language $\{w \in \{0,1\}^* \mid \text{every odd position of } w \text{ is a } 1\}$. Formulate an induction hypothesis, prove the base case, and show the induction step.

4. **(30 points)** Use the pumping lemma and (possibly) the closure properties of regular languages to prove that the following languages are non-regular:

   (a) $\{www \mid w \in \{0,1\}^*\}$

   (b) $\{0^n 1^m 0^n \mid m, n \geq 0\}$

   (c) $\{w \mid w \in \{0,1\}^*$ and is *not* a palindrome$\}$. Note that a string is a palindrome if it reads the same forward and backward.