Problem Set 4

Spring 10

Due: Thursday Mar 4 in class before the lecture.

Please follow the homework format guidelines posted on the class web page:

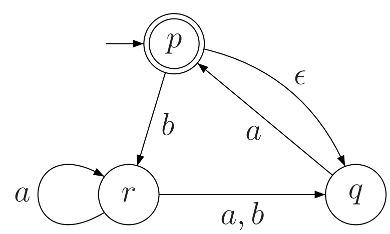
`http://www.cs.uiuc.edu/class/sp10/cs373/`

1. Regular expressions [**Category**: Comprehension, **Points**: 20]

   Construct an NFA with the same language as the regular expression $(a+b)^*(ab+b)$.

2. Regular expressions [**Category**: Comprehension, **Points**: 20]

   Use the algorithm learned in class to obtain a regular expression with the same language as the NFA below.



3. NFA and non-determinism [**Category**: Construction, **Points**: 20]

   Given $k > 0$, construct an NFA with at most $7k^2$ states for the following language over alphabet $\{0, 1, 2\}$

   $$L_k = \{ww' : |w| = |w'| = k, w \neq w'\}.$$

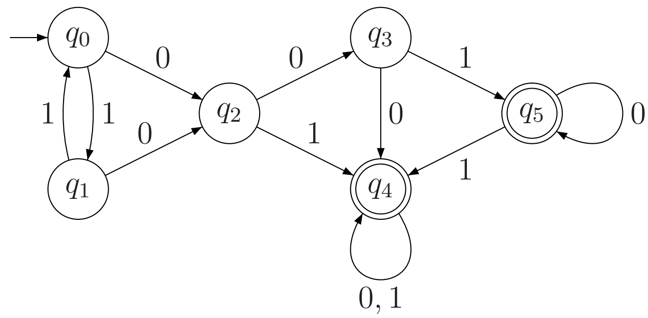
   (Hint: you need to exploit non-determinism to obtain such a small NFA).

4. Suffix languages [**Category**: Comprehension, **Points**: 20]

   (a) Give the suffix languages for the following DFA.

   (b) Merge states with the same suffix language to get a smaller DFA. Draw the resulting DFA.

   (c) Prove that all your suffix languages are different: for each pair of suffix languages give a sample string that belong to one languages and does not belong to the other one.



5. Non-regularity [**Category**: Proof., **Points**: 20]

   Prove that the following language is non-regular.
   $L = \{a^{4n}b^{3n} | n >= 0\}, \Sigma = \{a, b\}$

6. Concatenation [**Category**: EXTRA CREDIT, **Points**: 20]

   Due: Thursday Mar 11 in class before the lecture.

   In this question, we want to show that the smallest DFA accepting the concatenation of two languages L_1 and L_2 may be exponentially larger than the sizes of the DFAs for L_1 and L_2. Recall that when doing closure under concatenation, the construction in class was done for NFAs; we can do the same construction on DFAs, but the resulting automaton will be an NFA, and converting it to a DFA will give exponentially many states. We want to show that this exponential blow-up is unavoidable.

   To this end, let us fix $\Sigma = \{a, b\}$. Show that for any $k \in \mathbb{N}$, there exist two DFAs $A_1 = (Q_1, \Sigma, \delta_1, q_1^s, F_1)$ and $A_2 = (Q_2, \Sigma, \delta_2, q_2^s, F_2)$ with $O(k)$ states each, such that any DFA accepting $L(A_1)L(A_2)$ has at least $O(2^k)$ states. That is, any DFA accepting the concatenation of the languages of A_1 and A_2 will require states exponential in k.