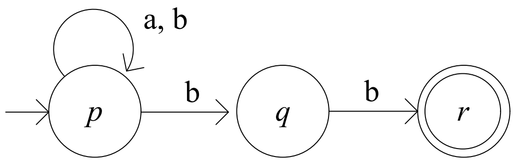
Quiz 4 Solutions

(1) Write a **regular expression** for the language of the following NFA M_1. Make it as simple as possible. Use standard abbreviations, not writing the concatenation symbol or extra parentheses.



$\boxed{(a \cup b)^* bb}$

(2) Suppose L is accepted by a 10-state **DFA** M_2. Using constructions described in class, we could convert M_2 into a $\boxed{20}$ -state **NFA** M_2' for $L\overline{L}$ (the bar denoting the complement of the language). We could then convert M_2' into a $\boxed{2^{20}}$ -state **DFA** for $L\overline{L}$.

(3) Darken the box to indicate if the statement is True or False. Really make your mark **dark**. *As always, a statement is True if it is always true; otherwise it is False.*

True Every regular language can be accepted by a DFA with an odd number of states.

True Every regular language can be accepted by a DFA whose start state is never visited twice.

Note: these were both taken from PS2.

(4) Same instructions. Throughout, fix an NFA $M = (Q, \Sigma, \delta, q_0, F)$.

True Suppose there is a $q_0 \rightsquigarrow q$ path in the diagram for M where $q \in F$ and the concatenation of edge-labels along the path is s. Then M accepts s.

False Suppose there is a $q_0 \rightsquigarrow q$ path in the diagram for M where $q \notin F$ and the concatenation of edge-labels along the path is s. Then M rejects s.

(5) Let's begin a proof that $L = \{a^n b^n : n \geq 0\}$ is **not** regular:

Assume for contradiction that L is regular. Then there is a DFA $M = (Q, \Sigma, \delta, q_0, F)$ that accepts L. Let $N = |Q|$. Consider the $N + 1$ strings $\boxed{\varepsilon, a, aa, \ldots, a^N}$. Each of these strings w determines a state $\delta^*(q_0, w)$. By the pigeonhole principle (PHP), we know that some two of these states $\boxed{\text{are the same)}}$. *And so on …*

Make sure that whatever you write gives grammatical English.