Regular Languages and Finite Automata

(a) Suppose that $L_1$ and $L_2$ are regular languages (over the same alphabet $\Sigma$) accepted by deterministic finite automata $M_1$ and $M_2$ respectively. Show that there is a deterministic finite automaton $M$ such that for all strings $u$ over $\Sigma$, $M$ accepts $u$ if and only if $u \notin L_1$ or $u \in L_2$. \[8 \text{ marks}\]

(b) Show that if a deterministic finite automaton $M$ over alphabet $\Sigma$ accepts all strings of length less than the number of states in $M$, then it must accept all strings over $\Sigma$. \[4 \text{ marks}\]

(c) What does it mean for two regular expressions over an alphabet $\Sigma$ to be equivalent? Using parts (a) and (b), or otherwise, describe an algorithm for deciding equivalence of regular expressions. State carefully any standard results that you rely upon. \[8 \text{ marks}\]