1) Let $N$ be an arbitrary NFA $N = \langle Q, \Sigma, \delta, q_0, F \rangle$. We will show how to create another NFA $N'$ with only one accept state such that $L(N) = L(N')$.

$N' = \langle Q', \Sigma, \delta', q_0', \delta_{\text{final}}, F' \rangle$ with

$Q' = Q \cup \epsilon_{\text{final}}$,

$\delta' = \delta \cup \epsilon_{\text{final}} \mid q_i \in F'$,

In essence, we add one new state to $N$, $q_{\text{final}}$, and make it an accept state. Then we add epsilon transitions from each of the original final states in $N$ to $q_{\text{final}}$ and leave this as the only accept state.

Now, we show $L(N) = L(N')$. Consider an arbitrary string that is accepted by $N$. There must exist a path for it to an accept state in $N$. From there, just take the epsilon transition to $q_{\text{final}}$. Thus $N'$ also accepts this string. Next consider any string accepted by $N'$. To get to its accept state, the string must have come directly from a state $q_i \in F$ on an epsilon state transition, meaning that $N$ accepts the string. Thus the languages are equal as desired.
2) \( L_1: \)

\[ q_0 \rightarrow q_1 \rightarrow q_2 \]

\( q_1: \) all strings that start with 1.

\( q_2: \) all strings that start with 1 and end in 0.

Designing an NFA for \( L_1 \) is a bit easier than a DFA since we don't need to define \( \delta(q_0, 0) \) and \( \delta(q_2, x), x \in \{0, 1\} \).

\( L_2: \)

\[ q_0 \rightarrow q_1 \rightarrow q_2 \rightarrow q_3 \rightarrow q_4 \]

\( q_1: \) ends in 0, \( q_2: \) ends in 01

\( q_3: \) ends in 010, \( q_4: \) ends in 0101.

Non-determinism helps here greatly by allowing for us to "guess" the last 4 digits of the string.

3) \( a \)

\( b \)
Let $D$ be a DFA that accepts $A$. We will use $D$ to create an NFA $A^R$ that accepts $A^R$. Make $N$'s accept state equal to $D$'s start state. For each transition in $D$ of the form $\delta(q_i, a) = q_j$, create a transition in $N$ of the form $\delta(q_j, a) = q_i$.

This "reverses" each transition. Finally, add a new start state to $N$, $q_0$. This state will have $\varepsilon$ transitions of the form $\delta(q_0, \varepsilon) = q_i$ for each $q_i \in F$, where $F$ is the set of final states in $D$. (Note: the only final state in $N$ will be the old start state of $D$, so none of the $q_i \in F$ in $D$ will be accept states in $N$ unless $q_0 \in F$.)

The rationale behind this transformation is to allow each string from the original language to be read in backwards.

Here is an example of the transformation:

Original DFA for $A$:

![Original DFA diagram]

NFA for $A^R$:

![NFA diagram]

Any sequence of states followed in the original DFA can be "reversed" in the NFA, after the appropriate $\varepsilon$ transition is taken.