Homework 1 Solutions

9.1. \((q_0, \Delta aaba) \rightarrow (q_1, \Delta_\bar{a}aba) \rightarrow (q_2, \Delta aaba) \rightarrow (q_2, \Delta Aaba) \rightarrow (q_4, \Delta Aaba) \rightarrow (q_4, \Delta Aaaba) \rightarrow (q_3, \Delta Aaba) \rightarrow (q_4, \Delta Aaba) \rightarrow (q_4, \Delta Aaaba) \rightarrow (q_3, \Delta Aaaba) \rightarrow (q_4, \Delta AABA) \rightarrow (q_5, \Delta AABA) \rightarrow (q_5, \Delta aABA) \rightarrow (q_6, \Delta aABA) \rightarrow (q_6, \Delta AABA) \rightarrow (q_8, \Delta AABA) \rightarrow (hr, \Delta AABA) \)
9.9. \( L(T_1) = L_1, \ L(T_2) = L_2 \), we want to build \( T \) d.e. \( L(T) = L_1L_2 \).

\[ \]

**Method One:** Add a transition from final states of \( T_1 \) to start state of \( T_2 \). This won't work: \( T_1 \) and \( T_2 \) expect an input string in their respective language, but \( w \in L_1L_2 \) may be such that \( w \notin L_1 \) and \( w \notin L_2 \). We must split \( w \) into \( xy \) such that \( w = xy \) and \( x \in L_1 \) and \( y \in L_2 \).

**Method Two:** Try every possible split \( w = xy \) one at a time. This won't work either. It may be that \( x \in L_1 \) and \( y \in L_2 \), but some prefix of \( x \) causes \( T_1 \) to go into an infinite loop.

**Method Three:** Try every possible split \( w = xy \) in parallel. That is, simultaneously simulate \( T_1 \) and \( T_2 \) on every possible value of \( x \) and \( y \), iteratively advancing each simulation by one step. This works, but it is hard to do!

**Method Four:** Nondeterministically choose \( x, y \) d.e. \( w = xy \); run \( T_1 \) on \( x \) and \( T_2 \) on \( y \) and accept iff both machines accept. This works.

9.1b. \( f(x) = a^n b^m \) where \( n \) is the number of 'a's in \( x \) and \( m \) is the number of 'b's.
9.20. **Modify**

1. Blanket tape with # for clean up.
2. Add clean-up and write '1' in place of $a_n$.
3. Add clean-up and write '0' in place of all crashes.

9.24. We begin in the state $(a_0, x, y)$ — i.e. anywhere on the tape. Note that if there are more than one 'a' or no 'a's the behavior is unspecified.

We also assume the input alphabet is $\{a, ?\}$ — then don't specify.

The TM does a scan from left to right over an ever-increasing portion of the tape. If it sees an 'a' it halts and accepts.