#1 Equivalency of DFAs

Consider the following language:

$$\mathsf{EQ}_{\mathsf{DFA}} = \{\, (D_1, D_2) \mid D_1 \text{ and } D_2 \text{ are equivalent DFAs} \,\}$$

Recall that two DFAs are equivalent if their languages are identical. Theorem 4.5 of Sipser shows that $\mathsf{EQ}_{\mathsf{DFA}}$ is decidable by way of language closure properties. In this problem, we'll consider a more direct approach.

1. Consider the following proof that $\mathsf{EQ}_{\mathsf{DFA}}$ is decidable by way of constructing a deciding Turing machine that recognizes the language.

   Proof. Define M, a deciding Turing machine that decides $\mathsf{EQ}_{\mathsf{DFA}}$ as follows.

   (a) M has inputs $\langle D_1, D_2 \rangle$ for DFAs D_1 and D_2.
   (b) For every possible string $w \in \Sigma^*$, if D_1 and D_2 have differing acceptance behavior on w, i.e., one rejects and the other accepts, then reject.
   (c) Otherwise, D_1 and D_2 have the same acceptance behavior on all strings: accept.

   □

   This construction has a fatal flaw! In a few sentences, describe what the flaw is.

2. We can patch up this flaw by noting that we don't have to test all strings. Determine a bound on the size of strings we need to test to determine if two DFAs are equal. In a few sentences, argue why this bound is correct.

   Hint: Simulate the input string w on D_1 and D_2 in a step-by-step, pairwise fashion. How many different state pairs are possible? Use the pigeon-hole principle to argue that strings over a certain length will necessarily incur a repetition.

#2 Prefix-free

1. Call a language L prefix-free if for all $w \in L$, there does not exist a $w' \in L$ such that $w \neq w'$ and w' is a prefix of w. For example ab is a prefix of $abcde$ and thus $abcde$ would not be in a prefix-free language if ab was also in the language. Give a deciding procedure to determine if the language of a DFA D is prefix free. Argue the correctness of your construction in a few sentences.