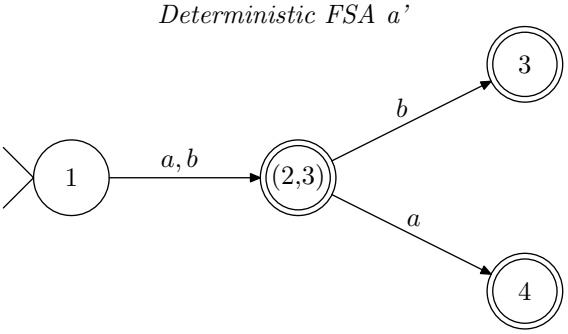
1 Questions 1 and 2

1.1

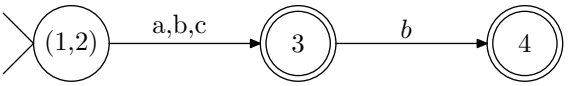
Deterministic FSA a'



Strings accepted by a (from the assignment):
a,b,aa,ab,ba,bb
Strings accepted by a':
a,b,aa,ab,ba,bb

1.2

Deterministic FSA b'



Strings accepted by b (from the assignment):
a,b,c,ab,bb,cb
Strings accepted by b':
a,b,c,ab,bb,cb

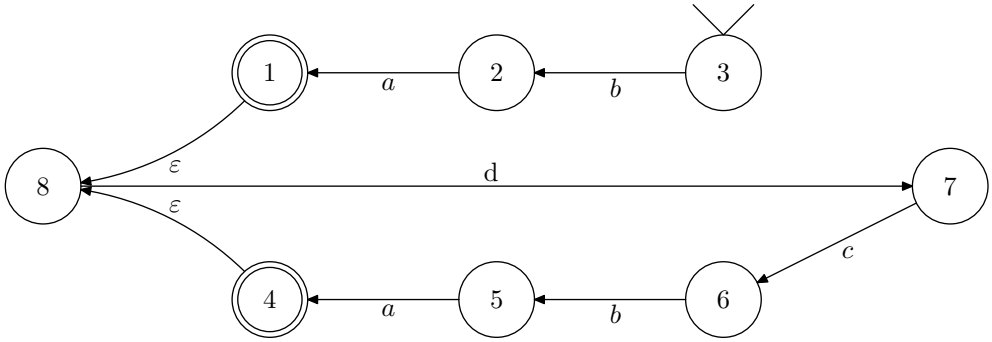
2 Question 3

2.1 We'll call the language that this NDFSA accepts L1. What are the four shortest strings in that language?

ab,abcdab,abcdabcdab,abcdabcdabcdab

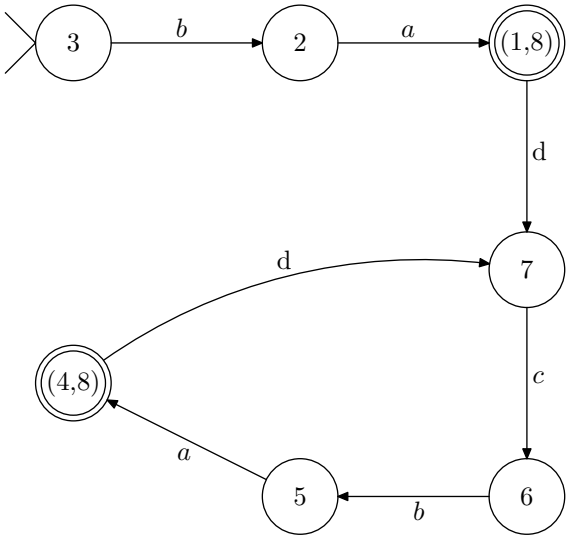
2.2

State diagram for L1r



The 4 shortest strings in L1r are:
ba,badcba,badcbadcba,badcbadcbadcba
These are the correct reverses of L1

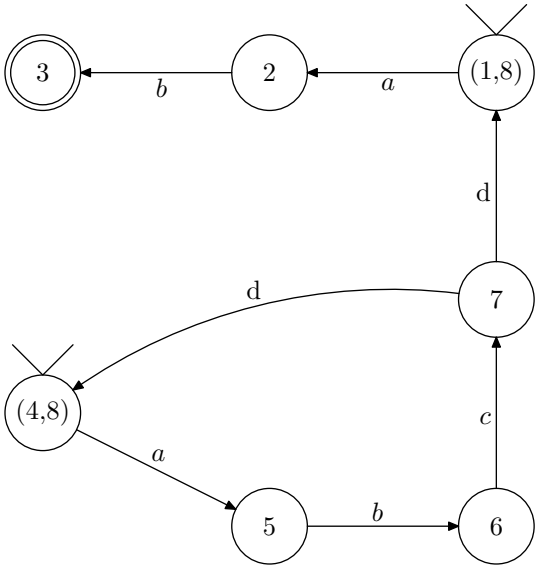
2.3 c.) A deterministic automaton for L1r:



This state machine has 7 states, one start state, and two end states (although you could easily collapse both end states into one state and compact the whole machine. That would diverge from the method outlined in the video however)

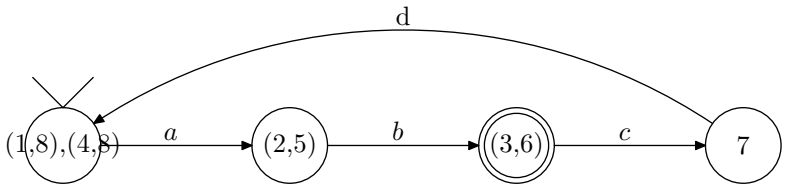
2.4

The reversed version of the previous machine



This is nondeterministic because it has two start states. Also from state 7 the letter d can take you to two other distinct states.

2.5 The deterministic version of the previous machine



This is different than the original machine because it has fewer total states. It only has 4. I don't think one with fewer states is possible because the language needs 4 specific characters in a specific sequence, which you need minimum 4 states to encode.