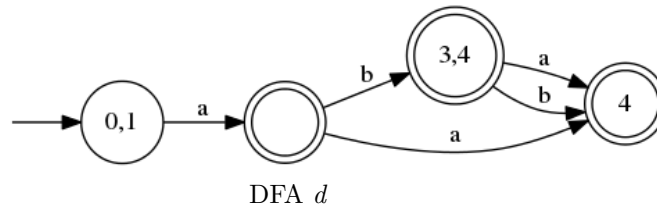
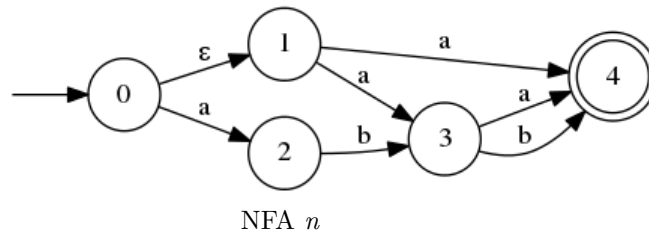


Assignment 1 Discussion

Problem 1: convert the NFA n to the DFA d via subset construction.



How to do this

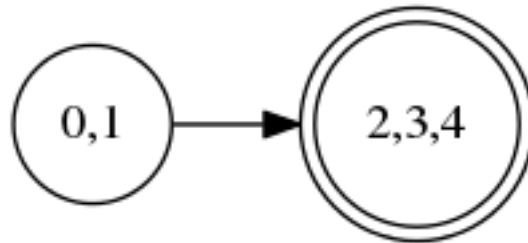
The first step is to create the ϵ -closure of the start state; that is we take all of the ϵ -transitions from the start state and merge the states into a single new state:



ϵ -closure of the start state

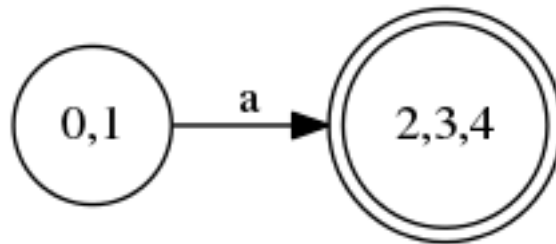
The next step is to create new states based on the possible moves from our start state node (and note that the new state created is also an accepting state)

since it includes the original state 4):



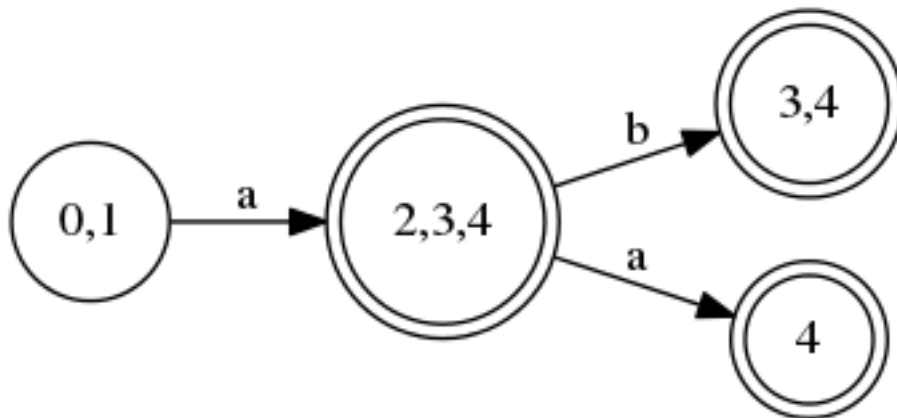
ϵ -closure of the start state

Now we take ϵ -closure of our new state, but since there are no epsilon transitions, we end up with the same diagram:



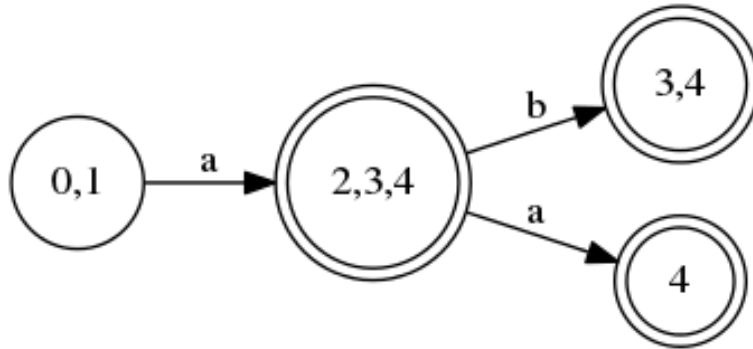
ϵ -closure of the start state

Now we again add the possible moves from our new state:



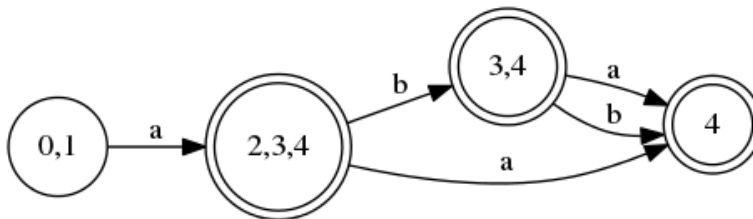
ϵ -closure of the start state

Again, since there are no ϵ -transitions from the new states, the DFA doesn't change:



ϵ -closure of the start state

Now we again add the possible moves from our new states; however, we note that the moves go only to a state identical to one that we have already created, so we merge that state and thus the new moves go to an already existing state:



ϵ -closure of the start state

Again, there are no epsilon transitions, and we since we didn't create any new states in this round, we are finished.

Problem 2

This is x86_64 assembly language (look at the reference to `r` versions of the registers and the use of `syscall`, which are not available in 32bit Intel instruction sets.)

Problem 3

This is Erlang (which you could also find by checking to see what `escript` might refer to.)

Problem 4

This is Haskell (several clues here, including use of the `IO()` monad – see https://wiki.haskell.org/IO_inside if you are very curious about this).

Problem 5

This is C.

Problem 6

The use of the Cambridge prefix syntax and `gosh` should be strong hints that this is Scheme.

Problem 7

The use of the initial `program` structure and `writeln` are strong clues that this is Pascal.

Problem 8

This is MIX assembly language; the process of elimination, if nothing else, should be a clue to this.