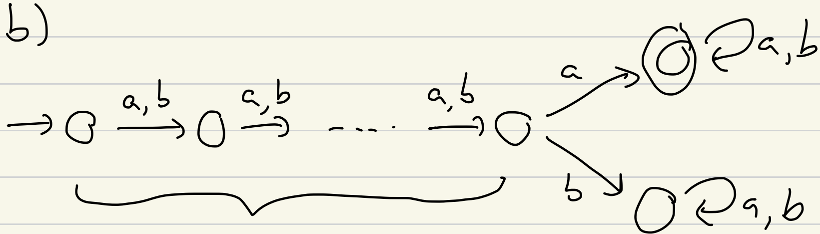
CPSC 421/501 Homework 7 Solutions 2023

Individual

(1) (a) Read the first $k-1$ symbols of the input, ignoring their values, and accept if the k^{th} symbol is a, reject otherwise

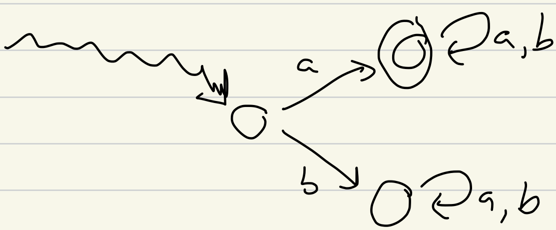
(b)

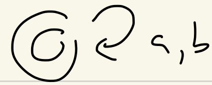


$k-1$ edges, or
k states in total

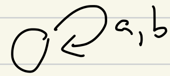
Explanation: The path of length $k-1$ reads and ignores the first $k-1$ symbols, the transitions, landing us in

the state



from here an "a" takes us to 

which <u>accepts</u> regardless of what follows,

and a "b" takes us to 

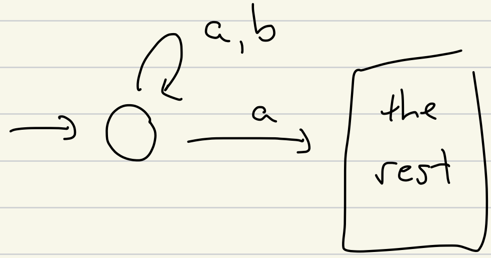
,
which (similarly) <u>rejects</u> regardless of what follows.

Note: The above descriptions may be a bit long-winded, but you must have some descriptions in parts (a) and (b) that go beyond mere descriptions of the DFA itself.

Same for parts (c) and (d) below.

(c) We allow ourselves to pass over any number of symbols in the input. At some we (non-deterministically) allow a symbol of "a" to take us to a part of the algorithm that ignores the next k-1 symbols and accepts (but doesn't allow for any more symbols of input).

(d) We start with

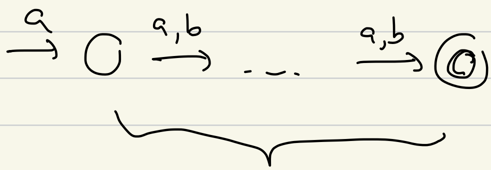


which allows us to read any number of symbols before the "a" transition.

From $\xrightarrow{a} \boxed{\text{the rest}}$ we read any

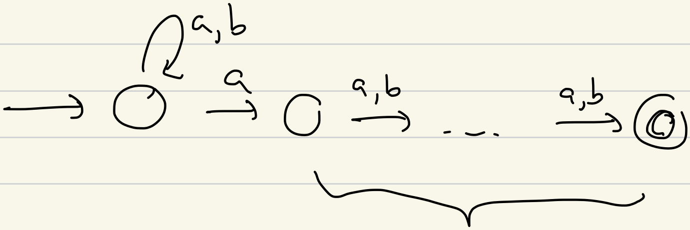
$k-1$ symbols and accept (rejecting anything longer). Hence

$\xrightarrow{a} \boxed{\text{the rest}}$ can be implemented by



path of length $k-1$.

So the entire NFA is



path of length $k-1$.

Note: In part (d), we used a slightly different style of description, which

- first describes the components,
- then gives the NFA,

rather than vice versa.

Either style is fine.

(2) $\text{AccFut}_{C_1}(\varepsilon) = C_1 = \Sigma^* a$

$\text{AccFut}_{C_1}(a) = \varepsilon \cup \Sigma^* a$

[Note: If you did the group homework first, you are more likely to write

$\text{AccFut}_{C_1}(a) = \varepsilon \cup \Sigma^* a$

$\text{AccFut}_{C_1}(b) = \Sigma^* a$,

and, sure, $\text{AccFut}_{C_1}(b) = \text{AccFut}_{C_1}(\varepsilon)$]

Note, here we use regular expressions like $\Sigma^* a$ and

$\varepsilon \cup \Sigma^* a$ since they are more convenient (perhaps....).

In an exam you are free to do the same.

In any case since one set contains ε and the other doesn't; these <u>two</u> sets are different.

Hence any DFA recognizing C_1 must have at least <u>two</u> states.

(3) Solution to be released later (but before the midterm).

Group Homework 7:

(1) For any $k \in \mathbb{N}$,

$$a^k \in \mathrm{AccFut}_L(a^k b)$$

since $a^k b a^k$ is a palindrome, and no shorter string lies in $\mathrm{AccFut}_L(a^k b)$:

indeed, if $\sigma_{k+2} \ldots \sigma_n \in \mathrm{AccFut}_L(a^k b)$

with $n < k$, and $\sigma_1 \ldots \sigma_n$ is a palindrome, then

$$\sigma_n, \sigma_{n-1}, \ldots, \sigma_{n-k+1} = a$$

and

$$\sigma_{n-k} = b$$

then $\sigma_{n-k} = b \neq a = \sigma_1 = \ldots = \sigma_k$

so $n-k > k$ and $n > 2k$ so $n \geq 2k+1$

Hence

$$|\sigma_{k+2}\cdots\sigma_n| = n-(k+2)+1$$
$$= n-k-1$$

satisfies

$$n-k-1 \geq (2k+1)-k-1 \geq k.$$

Since the shortest string in

$$\mathrm{AccFut}_L(a^k b)$$

is of length k, so for $k=1,2,3,\ldots$

all these sets are distinct.

Hence L is non-regular.

2 (a) Since $aaa\varepsilon = aaa \in C_3$,

but $baa\varepsilon = baa \notin C_2$, we have

$$\varepsilon \in \mathrm{AccFut}_{C_3}(aaa) \text{ and } \varepsilon \notin \mathrm{AccFut}_{C_3}(aaa)$$

(b) Similarly, for any $\sigma_1, \dots, \sigma_4$ we have

$$\varepsilon \in \mathrm{AccFut}_{C_3}(a\sigma_1\sigma_2), \quad \varepsilon \notin \mathrm{AccFut}_{C_3}(b\sigma_3\sigma_4).$$

(c) $\mathrm{AccFut}_{C_3}(a) = \Sigma^2 \cup C_3$,

since $at \in C_3$ implies $|t| \geq 2$,

so t can be any length 2 string,

or any element of C_3 (if $|t| \geq 3$,

then the "a" at the beginning of "at"

no effect on whether or not $at \in C_3$)

We also have

$$\mathrm{AccFut}_{C_3}(bba) = \Sigma^2 \cup C_3,$$

since the starting "bb" shows that $\mathrm{AccFut}_{C_3}(bba)$ contains no word of length ≤ 2. The "a" in "bba" shows that all strings of length 2 lie in $\mathrm{AccFut}_{C_3}(bba)$. Of course, for any string s we have

$$C_3 \subset \mathrm{AccFut}_{C_3}(s).$$

(d) $\mathrm{AccFut}_{C_3}(\varepsilon) = C_3$, and

$\mathrm{AccFut}_{C_3}(bbb) = C_3$, since the starting "bbb" does not allow any strings of length ≤ 3 to lie in $\mathrm{AccFut}_{C_3}(bbb)$.

(e)(i) Let $s, s' \in \Sigma^3$. Parts (a,b) show that $\text{Accfut}_{C_3}(s) \neq \text{Accfut}_{C_3}(s')$ provided that the first letter of s and s' are different. Similarly, for any $\sigma_1, \ldots \sigma_4 \in \Sigma$ we have

$$b \in \text{Accfut}_{C_3}(\sigma_1 a \sigma_2), \quad b \notin \text{Accfut}_{C_3}(\sigma_3 b \sigma_4).$$

$$\left(\begin{array}{l} \text{ALSO} \\ a \in \text{Accfut}_{C_3}(\sigma_1 a \sigma_2), \quad a \notin \text{Accfut}_{C_3}(\sigma_3 b \sigma_4). \end{array} \right)$$

Hence $\text{Accfut}_{C_3}(s) \neq \text{Accfut}_{C_3}(s')$ provided that the second letter of s and s' are different.

Similarly $\Sigma^2 \subset \text{Accfut}_{C_3}(\sigma_1 \sigma_2 a)$

and Σ^2 has no string in

$$\mathrm{AccFut}_{C_3}(\sigma_3 \sigma_4 b).$$

Hence $$\mathrm{AccFut}_{C_3}(s) \neq \mathrm{AccFut}_{C_3}(s')$$

provided that s and s' differ on their first, second, or third letter (symbol).

Since $s \neq s'$ implies that s, s' differ on (at least) one of their 1^{st}, 2^{nd}, or 3^{rd} letters, we have

$$\mathrm{AccFut}_{C_3}(s)$$

are different sets for the eight elements $s \in \Sigma^3$.

(e)(ii) If w is of length $n \geq 3$ and w ends in $\sigma_{n-2}\sigma_{n-1}\sigma_n \in \Sigma^3$

then

$$\mathrm{AccFut}_{C_3}(w) = \mathrm{AccFut}_{C_3}(\sigma_{n-2}\sigma_{n-1}\sigma_n),$$

since the first $n-3$ letters of w

irrelevant to whether or not

$wt \in C_3$

If $w = \sigma_1\sigma_2$ is of length 2,

then

$$\mathrm{AccFut}_{C_3}(\sigma_1\sigma_2) = \mathrm{AccFut}_{C_3}(b\sigma_1\sigma_2),$$

and similarly

$$\mathrm{AccFut}_{C_3}(\sigma_1) = \mathrm{AccFut}_{C_3}(bb\sigma_1)$$

and

$$\mathrm{AccFut}_{C_3}(\varepsilon) = \mathrm{AccFut}_{C_3}(bbb).$$

Hence there exactly 8 sets of

the form $\mathrm{AccFut}_{C_3}(w)$ as

w varies over all $w \in \Sigma^*$.

(3) [Similarly to (2) there are

4 possible values of

$$\mathrm{AccFut}_L(w)$$

as w varies over all $w \in \Sigma^*$, namely

$\mathrm{AccFut}_L(aa)$, $\mathrm{AccFut}(ab)$,
$\mathrm{AccFut}_L(ba)$, $\mathrm{AccFut}(bb)$.]

(3a) $\mathrm{AccFut}_{C_2}(bba) = \mathrm{AccFut}_{C_2}(ba)$,

hence
$\delta(q\,a)$ is the state of all
strings whose accepting future equals

$$\mathrm{AccFut}_{C_2}(ba)$$

OR, in brief:

$$\mathrm{AccFut}_{C_2}(\cdot) = \mathrm{AccFut}_{C_2}(ba)$$

in the notation in class.

(3b) Similarly

$\delta(q, a)$ = state corresponding to

$$\mathrm{AccFut}_{C_2}(\sigma_2 a)$$

and

$\delta(q, b)$ = " " "

$$\mathrm{Accfut}_{C}(\sigma_2 b)$$

(3c) $\mathrm{AccFut}_{C_2}(\varepsilon) = \mathrm{Accfut}_{C_2}(bb)$,

(and the initial state is where ε winds up in the DFA). Hence

q_0 = state corresponding to $\mathrm{AccFut}_{C_2}(bb)$.

(3d) A state corresponding to $\mathrm{AccFut}_{C_2}(s)$ is accepting iff $\varepsilon \in \mathrm{AccFut}_{C_2}(s)$. Since

$$aa, ab \in C_2, \qquad ba, bb \notin C_2$$

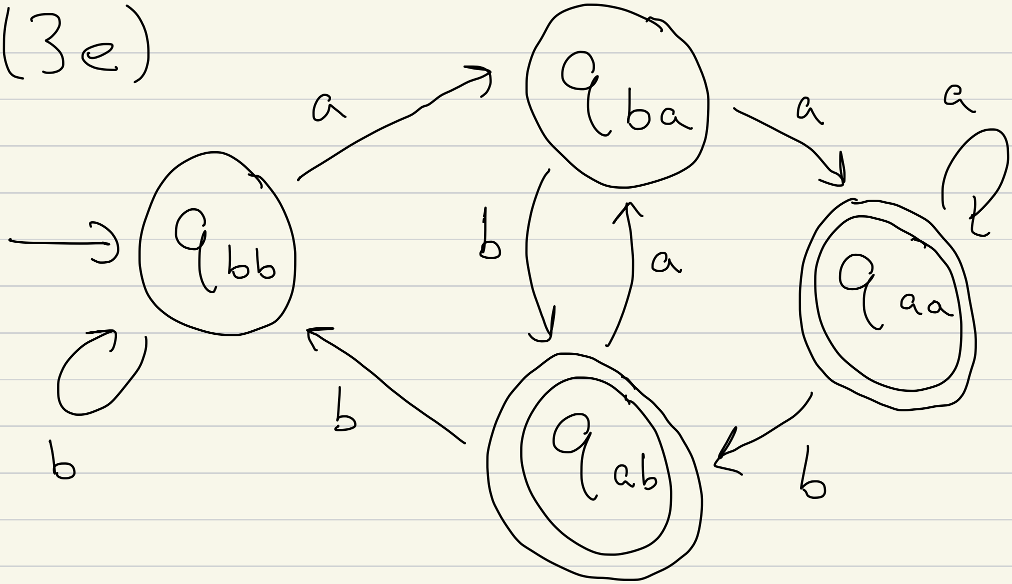
we have

$$\begin{array}{lcl} \varepsilon \in \mathrm{AccFut}_{C_2}(aa) & & \varepsilon \notin \mathrm{AccFut}_{C_2}(ba) \\ & \text{BUT} & \\ \varepsilon \in \mathrm{AccFut}_{C_2}(ab) & & \varepsilon \notin \mathrm{AccFut}_{C_2}(bb) \end{array}$$

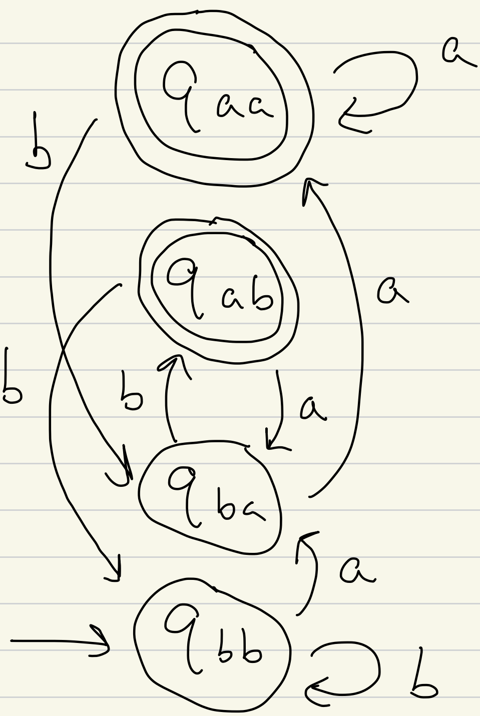
Hence F consists of the states corresponding to

$$\mathrm{AccFut}_{C_2}(aa), \quad \mathrm{AccFut}_{C_2}(ab).$$

(3e)



EQUIVALENTLY:



where

$$q_\omega = \text{state corresponding to } \mathrm{AccFut}_{C_2}(\omega).$$