Computer Science 181, Homework 3

Michael Wu
UID: 404751542

April 23rd, 2018

Postponed Problem 5

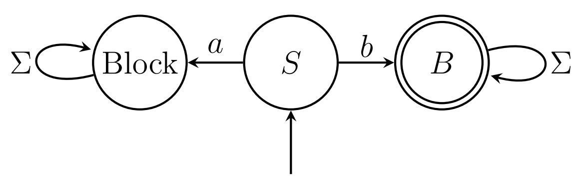
The set of finite state languages is closed under homomorphisms on a language's alphabet Σ. Therefore, if L_{five} is a finite state language, every homomorphism on $\Sigma = \{a, b, c\}$ results in a finite state language when applied to L_{five}. But the homomorphism

$$h(a) = a \qquad h(b) = b \qquad h(c) = \epsilon$$

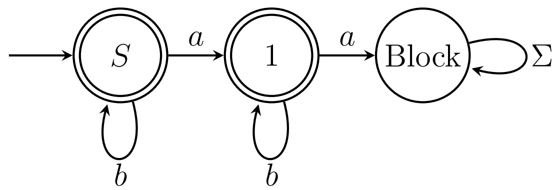
yields $h(L_{\text{five}}) = L_{\text{NFS}}$, which is not a finite state language. Therefore L_{five} cannot be a finite state language.

Problem 0

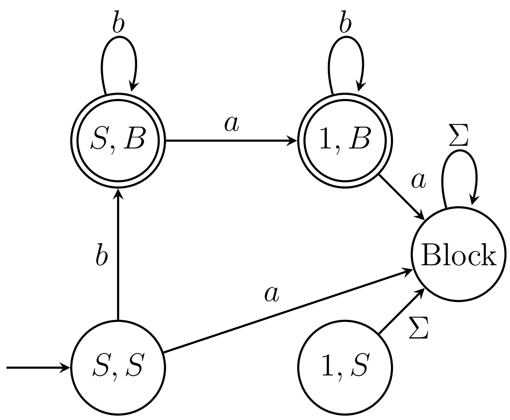
The first DFA can be made as shown below.



The second DFA can be made as shown below.



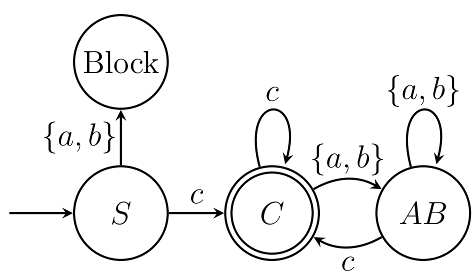
Their intersection should be as shown below.



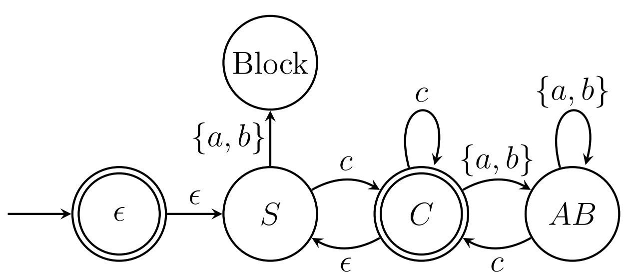
Here I have simplified a bit by grouping all blocking states into a single one, as in an intersection if any DFA blocks then their intersection will block. I also include the useless state $1, S$ to illustrate the cartesian product of the two DFAs.

Problem 1

a) A DFA for L_{one} is shown below.



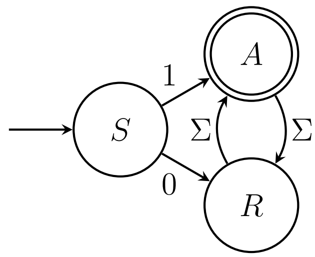
The corresponding NFA for L_{one}^* is shown below.



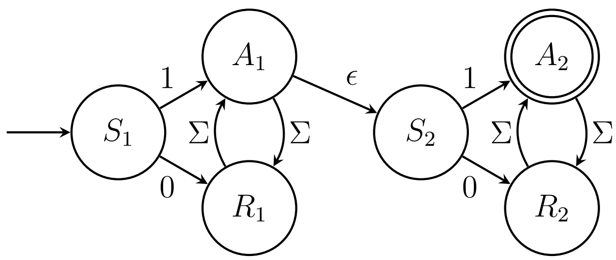
b) No they are not the same, as L_{one} does not contain the empty string ϵ, while L^*_{one} does.

Problem 2

a) A DFA for L_{two} is shown below.



The corresponding NFA for $L_{\text{two}} \cdot L_{\text{two}}$ is shown below.



b) $L_{\text{two}} \cdot L_{\text{two}}$ does not equal L_{two}, since 1 is in L_{two} but not in $L_{\text{two}} \cdot L_{\text{two}}$.

Problem 3

$$1000^* \cup 0100^* \cup 000^*10^* \cup 000^*$$

Problem 4

a)

Current State	Word	Next State
q_{start}	$w_1 = ab$	q_2
q_2	$w_2 = \epsilon$	q_1
q_1	$w_3 = ab$	q_1
q_1	$w_4 = aa$	q_2
q_2	$w_5 = ba$	q_{accept}

b) This is false, as the string $abab$ will be accepted. First ab will cause a move from q_{start} to q_2, then ab will cause a move into q_{accept}. This string has more than one b and an even number of a's, proving this statement is false.

Problem 5

Assume for contradiction that this language

$$L = \{0^i 1^j 0^k | i, j, k \geq 0 \land k = |i - j|\}$$

is finite state. Then by the pumping lemma if a language is finite state then there exists some p such that for any string s in L with $|s| > p$, s can be split into three strings x, y, and z where $s = xyz$, $|y| \geq 1$, $|xy| \leq p$, and $xy^*z \subseteq L$. Take the string $0^p 1^p \in L$, and note that xy must be a string made entirely of 0's, since $|xy| \leq p$. Additionally, y must be in 0^+, since $|y| \geq 1$. Let $x = 0^a$ and $y = 0^b$ for some constants $a \geq 0$ and $b > 0$. Then z must be $0^{p-a-b}1^p$. By the pumping lemma $xy^*z \subseteq L$, so

$$\{0^a(0^b)^*0^{p-a-b}1^p | a \geq 0 \land b > 0\} \subseteq L$$

This can be stated equivalently as

$$\{0^{p-b}0^{xb}1^p | \forall x \in \{0, 1, 2, \ldots\} \land b > 0\} \subseteq L$$

When $x = 2$, this means that the string $0^{p+b}1^p \in L$ for some constant $b > 0$. But in the definition of L, this string has the parameters $i = p + b$, $j = p$, and $k = 0$. Thus $k \neq |i - j|$ and $0^{p+b}1^p \notin L$. This is a contradiction, so L cannot be finite state.