



1. Webber Chap. 7 Exercise 1

Give a regular expression for each of the following languages.

- (a) **Language:** $\{abc\}$
Regular expression: abc
- (b) **Language:** $\{abc, xyz\}$
Regular expression: $abc+xyz$
- (c) **Language:** $\{a, b, c\}^*$
Regular expression: $(a+b+c)^*$
- (d) **Language:** $\{ax \mid x \in \{a, b\}^*\}$
Regular expression: $a(a+b)^*$
- (e) **Language:** $\{axb \mid x \in \{a, b\}^*\}$
Regular expression: $a(a+b)^*b$
- (f) **Language:** $\{(ab)^n\}$
Regular expression: $(ab)^*$
- (g) **Language:** $\{x \in \{a,b\}^* \mid x \text{ contains at least three consecutive as}\}$
Regular expression: $(a+b)^*aaa(a+b)^*$
- (h) **Language:** $\{x \in \{a,b\}^* \mid \text{the substring } bab \text{ occurs somewhere in } x\}$
Regular expression: $(a+b)^*bab(a+b)^*$
- (i) **Language:** $\{x \in \{a,b\}^* \mid x \text{ starts with at least three consecutive as}\}$
Regular expression: $aaa(a+b)^*$
- (j) **Language:** $\{x \in \{a,b\}^* \mid x \text{ ends with at least three consecutive as}\}$
Regular expression: $(a+b)^*aaa$

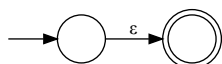
2. Webber Chap. 7 Exercise 2

For each of these regular expressions, give two NFAs: the exact one constructed by the proof of Lemma 7.1, and the smallest one you can think of.

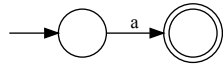
- (a) Exact and smallest NFA:



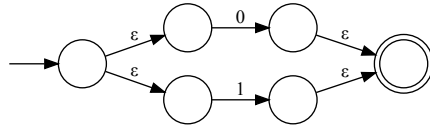
- (b) Exact and smallest NFA:



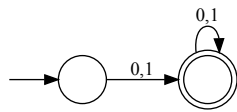
(c) Exact and smallest NFA:



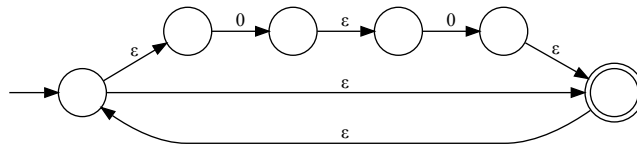
(d) Exact NFA



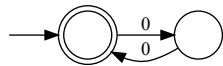
Smallest NFA:



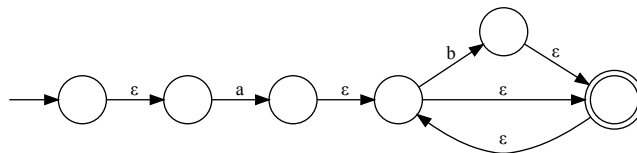
(e) Exact NFA



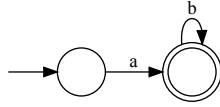
Smallest NFA:



(f) Exact NFA



Smallest NFA:



3. Webber Chap. 7 Exercise 3

For the following DFA, give a regular expression for each of the languages indicated. When the question refers to a machine “passing through” a given state, that means entering and then exiting the state. Merely starting in a state or ending in it does not count as “passing through.”

- (a) ϵ
- (b) b
- (c) b
- (d) a^*
- (e) a^*b
- (f) ba^*

4. Webber Chap. 8 Exercise 2

Show an egrep command that reads the standard input and echoes only those lines over the alphabet {a,b} that have an odd number of as.

```
egrep ^b*a(b*ab*a)*b*$ fileName.txt
```