

Do problems 0 and 1 and any two of 2, 3, or 4. Graded on a scale of 100 points.

0. **(5 points)** Your name: _____ Your student #: _____

1. **(35 points)** (Sipser exercise 1.47)

Let $\Sigma = \{1, \#\}$ and let

$$A = \{w \mid w = x_1\#x_2\#\cdots\#x_k, k \geq 0, \text{ each } x_i \in 1^* \text{ and } (i \neq j) \Rightarrow (x_i \neq x_j)\}$$

In English, A is the set of all strings consisting of zero or more strings of 1's separated by #'s such that no two of these strings of 1's have the same length. For example 1, 1#11#111, 1111##11#1111111 and 111#1111#11111#11111 are in A , but 1#1 and 1#11#111#11 are not.

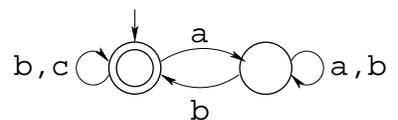
Prove that A is not regular.

2. (30 points)

- (a) (10 points) Give a DFA that recognizes the language $a(a \cup b)^*b \cup b(b \cup a)^*a$. The input alphabet is $\{a, b\}$. Drawing a state diagram for your DFA is sufficient.

- (b) (10 points) Give a NFA that recognizes the language $(ab^*)^*c \cup (ab)^*$. The input alphabet is $\{a, b, c\}$. Drawing a state diagram for your NFA is sufficient.

- (c) (10 points) Give a regular expression corresponding to the NFA:



Answer: _____

3. (35 points) Let B be any language. Define

$$f(B) = \{w \mid \exists x \in B. x = ww^{\mathcal{R}}\}$$

where $x^{\mathcal{R}}$ denotes the reverse of string x . For example,

$$f(\{\text{cattac, doggod, mouseesoum}\}) = \{\text{cat, dog, mouse}\}$$

Show that if B is any regular language, then $f(B)$ is regular as well. It is sufficient to describe the construction of a DFA, NFA or regular expression for $f(B)$ and/or use closure properties that we have already proven. You don't need to give a formal proof that your construction is correct.

4. **(35 points)** Ever had a broken keyboard that dropped or repeated characters? If so, this problem is for you. Let Σ be a finite alphabet, and let $RE(\Sigma)$ denote all regular expressions over strings in Σ^* . Define $flakeyKeys : \Sigma^* \rightarrow RE(\Sigma^*)$ as shown below

$$\begin{aligned}flakeyKeys(\epsilon) &= \epsilon \\flakeyKeys(x \cdot c) &= x \cdot c^*, \text{ for any } c \in \Sigma\end{aligned}$$

In other words, $flakeyKeys(x)$ maps the string x to a regular expression that matches any string that can be derived from x by dropping or repeating symbols. For example, $flakeyKeys(\text{cat})$ is the regular expression $c^*a^*t^*$

Let C be any language. Define

$$flakeyKeys(C) = \{w \mid \exists x \in C. w \in flakeyKeys(x)\}$$

Show that if C is regular, then $flakeyKeys(C)$ is regular as well. It is sufficient to describe the construction of a DFA, NFA or regular expression for $flakeyKeys(C)$ and/or use closure properties that we have already proven. You don't need to give a formal proof that your construction is correct.