CSC236 Homework Assignment #3

Language Regularity, Regular Expressions, and DFA/NFA Complexity

Alexander He Meng

Prepared for November 25, 2024

Question #1

Let $\Sigma \in \{0, 1\}$.

(a):

<u>Claim:</u> $\Sigma*$ is a regular language.

Proof.
proofgoeshere

□

(b):

<u>Claim:</u> $\Sigma * \backslash \{K\}, K = \{01, 101, 010\}$ is a regular language.

Proof.
proofgoeshere

□

(c):

<u>Claim:</u> $\{w|w$ is a palindrome$\}$ is NOT a regular language.

Proof.
proofgoeshere

□

(d):

<u>Claim:</u> $\{ww|w \in \Sigma*\}$ is NOT a regular language.

Proof.
proofgoeshere

□

(e):

Claim: $\{w|ww \in \Sigma*\}$ is a regular language.

Proof.
proofgoeshere

□

(f):

Claim: $\{w|w$ is a binary representation of a multiple of $3\}$ is a regular language.

Proof.
proofgoeshere

□

Question #2

Claim: Regular expressions that also have access to complement can still only express the same class of languages (i.e. the class of regular langauges) as regular expressions without the complement operation.

Proof.
proofgoeshere

□

Question #3

Counter-free languages are a subset of languages that satisfy the condition:

$$(\exists n \in \mathbb{N})(\forall m \geq n)(xy^m z \in L \iff xy^n z \in L).$$

Star-free regular expressions are regular expressions without the Kleene star, but with complementation.

It is known in formal language theory that counter-free languages are equivalent to the languages that can be expressed as **star-free regular expressions**.

(a):
<u>Claim:</u> $(ab)*$ can be matched with a star-free regular expression, where $\Sigma = \{a, b\}$.

Proof.
proofgoeshere

□

(b):
<u>Claim:</u> $(ab)*$ is not a counter-free language, where $\Sigma = \{a, b\}$.

Proof.
proofgoeshere

□

(c):
<u>Claim:</u> $(aa)*$ is not a counter-free language, where $\Sigma = \{a\}$.

Proof.
proofgoeshere

□

Question #4

Consider the language $L = \{w|\text{the third last character of } w \text{ is } 1\}$.

Let $k \in \mathbb{N}$ be arbitrary.

(a):

Claim: A DFA that accepts L has to have at least 2^k number of states.

Proof.
proofgoeshere

□

(b):

Claim: The smallest NFA that accepts L has to have exactly k number of states.

Proof.
proofgoeshere

□

(c):

Claim: The smallest DFA that accepts L has to have exactly $2^{k+1} - 1$ number of states.

Proof.
proofgoeshere

□

Question #5

<u>Claim:</u> Every finite language can be represented by a regular expression (meaning all finite languages are regular).

Proof.
proofgoeshere

□