• Don’t panic!

• You have 120 minutes to answer five questions. The questions are described in more detail in a separate handout.

• If you brought anything except your writing implements, your hand-written double-sided 8½" × 11" cheat sheet, and your university ID, please put it away for the duration of the exam. In particular, please turn off and put away all medically unnecessary electronic devices.

• Please clearly print your name and your NetID in the boxes above.

• Please also print your name at the top of every page of the answer booklet, except this cover page. We want to make sure that if a staple falls out, we can reassemble your answer booklet. (It doesn't happen often, but it does happen.)

• Do not write outside the black boxes on each page. These indicate the area of the page that our scanner can actually see. Anything you write outside the boxes will be erased before we start grading.

• If you run out of space for an answer, feel free to use the scratch pages at the back of the answer booklet, but please clearly indicate where we should look. Please ask for more scratch paper if you need it.

• Proofs or other justifications are required for full credit if and only if we explicitly ask for them, using the word prove or justify in bold italics.

• Please return all paper with your answer booklet: your question sheet, your cheat sheet, and all scratch paper. Please put all loose paper inside your answer booklet.
For each of the following languages \(L\) over the alphabet \(\Sigma = \{0, 1\}\), describe a DFA that accepts \(L\) and give a regular expression that represents \(L\). You do not need to justify your answers.

(a) All strings in which the number of runs is divisible by 3. (Recall that a run is a maximal non-empty substring where all symbols are equal.)

(b) All strings that do not contain the substring \(0110\).
Let $\text{take2skip2}(w)$ be the string function defined in the question handout, and let $L$ be an arbitrary regular language.

(a) **Prove** that the language $\{w \in \Sigma^* \mid \text{take2skip2}(w) \in L\}$ is regular.

(b) **Prove** that the language $\{\text{take2skip2}(w) \mid w \in L\}$ is regular.
Consider the following recursive function censor defined in the question handout.

(a) **Prove** that $|\text{censor}(w)| \leq |w|$ for all strings $w$.

(b) **Prove** that $\text{censor}(\text{censor}(w)) = \text{censor}(w)$ for all strings $w$.

As usual, you can assume any result proved in class, in the lecture notes, in labs, or in homework solutions.
Consider the language $L = \{a^b b^a \mid a > 2b \text{ or } 2a < b\}$.

(a) **Prove** that $L$ is not a regular language.

(b) Describe a context-free grammar for $L$. 

---
For each statement below, check “Yes” if the statement is *always* true and check “No” otherwise, and write a *brief* (one short sentence) explanation of your answer. Read these statements very carefully—small details matter!

(a) For every language $L$, the language $L^*$ is infinite.

$$
\begin{array}{c}
\text{Yes} \\
\text{No}
\end{array}
$$

(b) If a language $L$ is finite, the complement of $L$ is context-free.

$$
\begin{array}{c}
\text{Yes} \\
\text{No}
\end{array}
$$

(c) The language $\{0^{374n} \mid n \geq 374\}$ is regular.

$$
\begin{array}{c}
\text{Yes} \\
\text{No}
\end{array}
$$

(d) The language $\{wxw^R \mid w, x \in \Sigma^\ast\}$ is regular.

$$
\begin{array}{c}
\text{Yes} \\
\text{No}
\end{array}
$$

(e) The context-free grammar $S \rightarrow 0S1S \mid S1S0 \mid \epsilon$ generates the set of all binary strings with the same number of 0s and 1s.

$$
\begin{array}{c}
\text{Yes} \\
\text{No}
\end{array}
$$

(f) Every regular language is recognized by a DFA with at least 374 states.

$$
\begin{array}{c}
\text{Yes} \\
\text{No}
\end{array}
$$

(g) If the languages $L$ and $L'$ are regular, their intersection $L \cap L'$ is also regular.

$$
\begin{array}{c}
\text{Yes} \\
\text{No}
\end{array}
$$

(h) If a language has an infinite fooling set, then it is context-free.

$$
\begin{array}{c}
\text{Yes} \\
\text{No}
\end{array}
$$

(i) Let $M$ be a DFA over the alphabet $\Sigma$. Let $M'$ be identical to $M$, except that accepting states in $M$ are non-accepting in $M'$ and vice versa. Each string in $\Sigma^\ast$ is accepted by exactly one of $M$ and $M'$.

$$
\begin{array}{c}
\text{Yes} \\
\text{No}
\end{array}
$$

(j) Let $M$ be an NFA over the alphabet $\Sigma$. Let $M'$ be identical to $M$, except that accepting states in $M$ are non-accepting in $M'$ and vice versa. Each string in $\Sigma^\ast$ is accepted by exactly one of $M$ and $M'$.

$$
\begin{array}{c}
\text{Yes} \\
\text{No}
\end{array}
$$