CS Theory (Spring '25)	Assigned: February 25, 2025
Homework 3	
Instructor: *Josh Alman*	Due: March 6, 2025 at noon

Collaboration on homework is allowed. However, you must write up solutions by yourself and understand everything that you hand in. You may also consult other reference materials, but you may not seek out answers from other sources or use AI tools. See the course webpage for more details.

Write the solution to each part of each problem on a separate page. Be sure to correctly indicate which page each problem appears on in GradeScope. Please do not write your name on any page of the submission; we are using anonymous grading in GradeScope.

There are 100 total possible points, and also bonus questions worth an additional 30 points.

0 Exercises

Here are some recommended exercises. You should not turn in your solutions for these, and we will not grade them. From the Sipser textbook: Exercises 1.18, 1.20

Additional exercise 1: For each of the following languages, determine its streaming complexity by giving a big-O upper bound and a matching (up to the constant) lower bound. (Recall from class that a language has a $O(1)$ space streaming algorithm if and only if it is regular, so if your answer is $O(1)$, be sure you also think the language is regular!)

(a) $\{w \in \{0,1\}^* \mid \text{ the length of } w \text{ is a power of } 2\}$

(b) $\{w \in \{0,1\}^* \mid \text{ the length of } w \text{ is odd}\}$

Additional exercise 2: Determine whether each of the following statements is true or false.

(a) For all functions $f, g : \mathbb{N} \to \mathbb{N}$, if $f(n) = o(g(n))$ then $f(n) + 100 \cdot g(n) = O(g(n))$.

(b) $\log_2(n) = O(\log_3(n))$.

(c) $2^{2n} = O(2^n)$.

(d) $\log_2 \log_2(n) \cdot \sqrt{n} = O(\log_2(n) \cdot n^{1/4})$.

1 Starlessness (20 points; each part 10 points)

We say a regex is *starless* if it does not contain any $*$s. So, equivalently, it is only defined using the first 5 recursive rules defining a regex (as in class / page 64 of the textbook), and not the last one.

(a) Prove that the language corresponding to any starless regex is finite.

(b) For every positive integer n, give a starless regex over the alphabet $\Sigma = \{a, b\}$ of length at most $O(n)$ whose corresponding language has size at least 2^n. Use only the rules from the definition of a regex on page 64 of the textbook, and no other symbols or abbreviations. For instance, ab is *not* a valid regex for the language $\{ab\}$ using just these rules; you must apply rule 5 to two different regexes gotten from rule 1 to yield $(a \circ b)$, which has length 5. Please give an accompanying English explanation of why there are at least 2^n strings in the corresponding language.

2 Bounded by polynomials (10 points)

Recall from class that, if $f : \mathbb{N} \to \mathbb{N}$ is a function, we write $f = \text{poly}(n)$ to denote that there is a constant $c > 0$ such that $f(n) = O(n^c)$. This is equivalent to saying $f(n) = n^{O(1)}$, and also equivalent to saying $f(n) = 2^{O(\log n)}$. Prove that if $f : \mathbb{N} \to \mathbb{N}$ and $g : \mathbb{N} \to \mathbb{N}$ are such that $f(n) = \text{poly}(n)$ and $g(n) = \text{poly}(n)$, then $f(g(n)) = \text{poly}(n)$.

3 Defining Streaming with DIAs (20 points; each part 10 points)

Recall from homework 2 that a DIA is a DFA $(Q, \Sigma, q_0, F, \delta)$ where the sets Q, F (the set of states and the set of accept states) are allowed to be infinitely large. For every positive integer n, let's furthermore define $Q_n \subseteq Q$ to be the set of states that can be reached by an input string of length $\leq n$.

Recall from class that we defined streaming algorithms and their space usage. In this problem, we'll show that DIAs and Q_n can be used to capture essentially the same notion.

Prove that for every language L over $\Sigma = \{0, 1\}$, the following are true:

(a) If there is a streaming algorithm (using the definition from class) for L that uses $S(n)$ space, then there is a DIA for L such that, for all integers n, we have $|Q_n| = 2^{O(S(n))}$.

(b) If there is a DIA for L (with sets Q_n defined above), then there is a streaming algorithm (using the definition from class) which uses space $S(n) = O(\log_2(|Q_n|))$.

4 Balance (20 points; each part 10 points)

A string $w \in \{0, 1\}^*$ is called *balanced* if more than a quarter of its bits are 1 and more than a quarter of its bits are 0. Define the language $L = \{w \in \{0, 1\}^* \mid w \text{ is balanced}\}$.

(a) Give a streaming algorithm for L which uses at most $O(\log(n))$ space.

(b) Prove that every streaming algorithm for L requires $\geq \frac{1}{1000} \log_2(n)$ space for inputs of length $\leq n$.

5 Communication (20 points; each part 10 points)

(a) Prove that the communication complexity of the function $f : \{0,1\}^* \times \{0,1\}^* \to \{0,1\}$ defined below is $O(\log n)$.

$$f(x,y) = \begin{cases} 1 & \text{if } x \text{ has more 1s than } y, \\ 0 & \text{otherwise.} \end{cases}$$

(b) For strings $x, y \in \{0,1\}^*$, we say x and y *intersect* if there is an index i such that the ith character of x and the ith character of y are both 1. For instance, 00100 and 10101 intersect (with $i = 3$) but 11001 and 00110 do not intersect. Prove that the communication complexity of the function $g : \{0,1\}^* \times \{0,1\}^* \to \{0,1\}$ defined below is $O(\sqrt{n} \log n)$.

$$g(x,y) = \begin{cases} 1 & \text{if } x \text{ has more than } \sqrt{n} \text{ 1s,} \\ 1 & \text{if } y \text{ has more than } \sqrt{n} \text{ 1s,} \\ 1 & \text{if } x \text{ and } y \text{ intersect,} \\ 0 & \text{otherwise (if none of the above are true).} \end{cases}$$

6 BONUS Questions about Streaming (20 BONUS points; each part 10 BONUS points)

(a) Prove or disprove: If L is a language over $\Sigma = \{0,1\}$, and L has a streaming algorithm which uses at most $O(\log(\log n))$ space, then L is a regular language.

(b) Let $\Sigma = \{a, b, c\}$ and consider the language

$$L := \{wcw \mid w \in \{a,b\}^*\}.$$

Give (with proof) a language L' with the following three properties:

- $L \subseteq L'$,
- For all positive integers $n \geq 100$, we have $|L' \cap \Sigma^{2n+1}| \leq \frac{1}{3} 4^n$, and
- L' has a streaming algorithm with $O(\log n)$ space usage.

(One can prove that the streaming space complexity of L is $\Theta(n)$. No need to prove that here. Your streaming algorithm for L' can be thought of as a much more efficient streaming algorithm with "small one-sided error" for L: it accepts everything in L, and still rejects most "reasonably-formatted" strings that are not in L.)

7 Collaborators, References, and Formatting (10 points)

Please list who you collaborated with on each problem, including any TAs or students you discussed the problems with in office hours. Also list any reference materials consulted other than the lectures and textbook for our class. Finally, please follow the formatting guidelines at the top of this document.