CSE 105, Fall 2019 - Homework 1

Solutions

Instructions

Upload a single file to Gradescope for each group. All group members’ names and PIDs should be on each page of the submission. Your assignments in this class will be evaluated not only on the correctness of your answers, but on your ability to present your ideas clearly and logically. You should always explain how you arrived at your conclusions, using mathematically sound reasoning. Whether you use formal proof techniques or write a more informal argument for why something is true, your answers should always be well-supported. Your goal should be to convince the reader that your results and methods are sound.

Reading Sipser Chapter 0 and Section 1.1

Key Concepts Sets, integers, sequences, functions, relations, predicates, graphs, trees, strings, languages, lexicographic ordering, boolean logic, proof by construction, proof by contradiction, proof by induction, finite automata (DFA), computation trace, accept / reject, language of an automaton, regular language, union of languages, concatenation of languages, star of a language.
Problem 1 (10 points)

1. In this problem, we'll use the following definitions (from page 44) of operations on languages (sets of strings) A, B:

Union $A \cup B = \{ x \mid x \in A \text{ or } x \in B \}$
Concatenation $A \circ B = \{ xy \mid x \in A \text{ and } y \in B \}$
Star $A^* = \{ x_1x_2\ldots x_k \mid k \in \mathbb{Z} \text{ and } k \geq 0 \text{ and each } x_i \in A \}$

For each of the following sets of strings over the alphabet \{a, b\}, answer the following questions:

1. Is $\epsilon$ (the empty string) in the set?
2. What is an example of a string over \{a, b\} of length at least 2 that is in the set (or why isn’t there such an example)?
3. What is an example of a string over this alphabet of length at least 2 that is not in the set (or why isn’t there such an example)?

(a) \{a\}* $\cup$ \{b\]*
   (i) Yes, the Kleene star operator allows for zero instances of \{a,b\}.
   (ii) (aa) is in the set, as all strings consisting of characters just a and just b are in the set.
   (iii) (ab) is not in the set, as all strings consisting of characters just a and just b.

(b) \{a\} $\circ$ \{b\]*
   Note: \{a\} $\circ$ \{b\}* = a, ab, abb, abbb, ...
   (i) The empty string is not in the set, as every instance has to start with a.
   (ii) (ab) is in the set, as we start with a and it occurs when the Kleene star creates only one instance of b.
   (iii) (bb) is not in the set, as we cannot form it by repeating b any number of times since we always have to start with a.

(c) \{ab, ba\}*
   (i) Yes, the empty set is allowed (in any set of the form S*)
   (ii) A string in the set is (ab).
   (iii) A string not in the set is (aaa), as we cannot form it by any combination of (ab) and (ba).

(d) \{a, b\} $\cup$ \{aa, bb\}
   (i) No, because an empty string is not an element of either set that this set is a union of.
   (ii) A string in the set is (bb). It is in the second set, and thus is also in the union.
(iii) A string not in the set is (aaa). It is not in either of the sets that this set is a union of.

Problem 2 (10 points)

(a) What is the language recognized by M? Give an informal description in English and briefly justify your answer.

The language recognized by M is the set of all strings with an even number of 1’s and a number of 0’s that’s a multiple of three. This is because horizontally, the states represent the number of 0’s mod 3 and vertically, the states represent the number of 1’s mod 2 (so even or odd). The top left state thus represents both an even number of 1’s and a number of 0’s that’s a multiple of three (i.e. number of 1’s mod 3 = 0).

(b) Are there strings \( x \in L(M) \), such that if the bits of \( x \) are flipped (changing 0 to 1 and 1 to 0), the resulting string will also be in \( L(M) \)? Why or why not? Is this true for every string \( x \in L(M) \)? (Exclude the empty string)

Yes, there does exist such a string. For example, choose the string 11111000000. Since it has an even number of 1’s and two multiples of 000’s it is in the language \( L(M) \). If the bits are flipped, it still has an even number of 1’s and two multiples of 000’s, so it is still in the language.
This is not true for every string, because 11000 is in $L(M)$ but when the bits are flipped, we get 00111 which is not, since it neither has an even number of 1’s nor is the number of 000’s.

(c) Write the formal definition of $M = (Q, \Sigma, \delta, q_0, F)$. Use a table to define $\delta$.

**Common Mistake:** $F = q_0$. $F$ is a set of states.

$Q = \{q_0, q_1, q_2, q_3, q_4, q_5\}$
$\Sigma = \{0,1\}$
$F = \{q_0\}$
$q_0 = q_0$

$\delta$:  

<table>
<thead>
<tr>
<th>State</th>
<th>Input</th>
<th>Next state</th>
</tr>
</thead>
<tbody>
<tr>
<td>$q_0$</td>
<td>0</td>
<td>$q_1$</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>$q_3$</td>
</tr>
<tr>
<td>$q_1$</td>
<td>0</td>
<td>$q_2$</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>$q_4$</td>
</tr>
<tr>
<td>$q_2$</td>
<td>0</td>
<td>$q_0$</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>$q_5$</td>
</tr>
<tr>
<td>$q_3$</td>
<td>0</td>
<td>$q_4$</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>$q_0$</td>
</tr>
<tr>
<td>$q_4$</td>
<td>0</td>
<td>$q_5$</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>$q_1$</td>
</tr>
<tr>
<td>$q_5$</td>
<td>0</td>
<td>$q_3$</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>$q_2$</td>
</tr>
</tbody>
</table>
Problem 3 (10 points)

CSE 105 is a fairly proof heavy class, so the goal of this problem is to help refresh you on different proof techniques. In particular, proof by induction and closure proofs will be important in this class.

(a) Prove by induction that \(7^n - 2^n\) is divisible by 5, for all \(n \geq 0\)

Let \(S(n)\) denote the proposition that \(7^n - 2^n\) is divisible by 5, for all \(n \geq 0\).

\(S(0)\) is the statement that \(7^0 - 2^0 = 1 - 1 = 0\) is divisible by 5. Since every number other than zero divides zero, it follows that 5 divides 0 and \(S(0)\) is true.

Assume that \(S(k)\) is true, for some \(k > 0\), i.e., assume that \(7^k - 2^k\) is divisible by 5.

Accordingly, we can write, \(7^k - 2^k = 5 \cdot m\), where \(m\) is some integer.

Now we have to prove that \(S(k + 1)\) is true.

Observe that,

\[
7^{k+1} - 2^{k+1} = 7 \cdot 7^k - 2 \cdot 2^k
\]

\[
= 7 \cdot (2^k + 5 \cdot m) - 2 \cdot 2^k
\]

\[
= 7 \cdot 2^k + 7 \cdot (5 \cdot m) - 2 \cdot 2^k
\]

\[
= (7 - 2) \cdot 2^k + 5 \cdot (7 \cdot m)
\]

\[
= (5) \cdot 2^k + 5 \cdot (7 \cdot m)
\]

\[
= 5 \cdot q
\]

where \(q = 2^k + (7 \cdot m)\) is an integer, since \(k\) and \(m\) are integers. It follows that \(7^{k+1} - 2^{k+1}\) is divisible by 5, i.e., \(S(k + 1)\) is true.

(b) Are the odd integers closed under addition? multiplication? Show by proof.

Addition: We can see that \(3 + 5 = 8\). Because 3 and 5 are both odd but their sum isn’t, the odd integers are not closed under addition.

Multiplication: Let’s try multiplying some odds:

\[3 \times 5 = 15, \quad -7 \times 9 = -63, \quad -1 \times -7 = 7\]

Based on these three examples, it appears that the odd integers are perhaps closed under multiplication.
Proof: Note that every odd integer is of the form 2n+1 for some integer n. Let 2m + 1 and 2n + 1 be two arbitrary odd numbers. Then their product is given by:

\[(2m + 1)(2n + 1) = 4mn + 2m + 2n + 1 = 2(2mn + m + n) + 1\]

Since the integers are closed under multiplication and addition, 2mn + m + n is an integer. Thus \((2m+1)(2n+1)\) is an odd integer since its of the form 2S+1 for the integer \(S = 2mn + m + n\). This proves that the odd integers are closed under multiplication.

Common Mistake: The two integers cannot be taken as \((2m+1)\) for both. If you show that \((2m+1)(2m+1)\) is odd, all this proves is that the square of odd integers is odd. You need to be able to prove this property for the product of any 2 integers (Note: As in the solution, if m=n, it takes care of the case of squares. Hence, this is the more general solution).

Problem 4 (10 points)

\[(a)\] Write the formal definition of \(M = (Q, \Sigma, \delta, q_0, F)\). Use a table to define \(\delta\).

\[Q = \{q_0, q_1, q_2\}\]
\[\Sigma = \{a, b\}\]
\[F = \{q_0, q_1\}\]
\[q_0 = q_0\]
δ:

<table>
<thead>
<tr>
<th>State</th>
<th>Input</th>
<th>Next state</th>
</tr>
</thead>
<tbody>
<tr>
<td>q₀</td>
<td>a</td>
<td>q₀</td>
</tr>
<tr>
<td></td>
<td>b</td>
<td>q₁</td>
</tr>
<tr>
<td>q₁</td>
<td>a</td>
<td>q₂</td>
</tr>
<tr>
<td></td>
<td>b</td>
<td>q₁</td>
</tr>
<tr>
<td>q₂</td>
<td>a</td>
<td>q₂</td>
</tr>
<tr>
<td></td>
<td>b</td>
<td>q₂</td>
</tr>
</tbody>
</table>

(b) What is an example of a string $x$ such that both $x$ and $x^R$ ($x^R$ is the reverse of $x$) are in $L(M)$? (or why isn’t there such an example)?

Common Mistake: if $x = "aabbb"$, then $x^R = "bbbaa"$ not "bbaaa"

Empty string because this represents $a^*b^*$.

(c) What is an example of a string $y$ such that neither $y$ nor $y^R$ are in $L(M)$? (or why isn’t there such an example)?

(aba).

Problem 5 (10 points)

You have an incoming bitstream (sequence of 0s and 1s) that might be truncated at any time.

(a) Create a DFA (State diagram) that recognizes the language \{w | w contains an equal number of occurrences of the substrings 01 and 10\).
(b) Create a DFA (State diagram) to determine if the number represented (in binary, reading bits from left-to-right) is divisible by 3.

The first step in the design is to identify that the DFA must have 3 states: one state to denote strings that are exactly divisible by 3, one state to denote strings that result in a remainder of 1, and another state to denote strings that result in a remainder of 2, when divisible by 3.

The second observation is that appending a 0 to the right of a binary number causes its value as a number to double, whereas adding a 1 results in a number that is the sum of 1 and twice the original value.

The third set of observations are as follows:
(a) If \( p \equiv 0 \mod 3 \), then \( 2 \cdot p \equiv 0 \mod 3 \) and \( (2 \cdot p + 1) \equiv 1 \mod 3 \).
(b) If \( p \equiv 1 \mod 3 \), then \( 2 \cdot p \equiv 2 \mod 3 \) and \( (2 \cdot p + 1) \equiv 0 \mod 3 \).
(c) If \( p \equiv 2 \mod 3 \), then \( 2 \cdot p \equiv 1 \mod 3 \) and \( (2 \cdot p + 1) \equiv 2 \mod 3 \).

Common Mistake: The number represented by the string must be divisible by 3. This is not the same as having the number of 1’s in your string to be a multiple of 3!
(If you need to you can use http://madebyevan.com/fsm/ to draw your diagram.)