Dec 01, 2017
COMP.3040 Foundation of Computer Science
Viet Tran
vtran1@student.uml.edu
Homework V Solution

4.2 Consider the problem of determining whether a DFA and a regular expression are equivalent. Express this problem as a language and show that it is decidable.

To begin, have language $LM_{DFA,REX}$ = {⟨ Q,R⟩ | where Q is a DFA, R is a regular expression, and L(Q) = L(R)}. The following Turing Machine F will decide $LM_{DFA,REX}$.

F = Input $\langle Q, R\rangle$:

1. Take the regular expression R and convert it to an equivalent DFA S using Theorem 1.28.
2. Take TM C to decide LM_{DFA} from Theorem 4.5 for the input $\langle Q, S\rangle$.
3. If R accepts then accept, otherwise reject.

4.3 Let $ALL_DFA = \{\langle A\rangle|$ A is a DFA and L(A) = Σ*}. Show that ALL_{DFA} is decidable.

Consider $ALL_{DFA} = \{\langle A\rangle$ where A is a DFA and recognizes Σ*}. Turing machine M will determine ALL_{DFA}.

M = From input of $\langle A\rangle$, A being defined as a DFA:

1. Create a DFA C recognizing $\overline{L(A)}$ from Exercise 1.10.
2. Execute the Turing Machine defined in Theorem 4.4 on input $\langle B\rangle$ with the condition that T will determine E_{DFA}.
3. If T accepts then accept, otherwise reject.

4.4 Let $A\varepsilon_{CFG} = \{\langle G\rangle|$ G is a CFG that generates ε}. Show that $A\varepsilon_{CFG}$ is decidable.

Consider $A\varepsilon_{CFG} = \{\langle G\rangle$ where G is a CFG that decides ε}. Turing machine M will determine $A\varepsilon_{CFG}$.

M = From input of $\langle G\rangle$, G being defined as a CFG:

1. Execute Turing Machine S from Theorem 4.6 on the input $\langle G, \varepsilon \rangle$, and S will decide A_{CFG}.
2. If S accepts then accept, otherwise reject.

4.6 Let X be the set {1, 2, 3, 4, 5} and Y be the set {6, 7, 8, 9, 10}. We describe the functions $f : X \to Y$ and $g : X \to Y$ in the following tables. Answer each part and give a reason for each negative answer.

Table 1: Function f

n	$f(n)$
1	6
2	7
3	6
4	7
5	6

Table 2: Function g

n	$g(n)$
1	10
2	9
3	8
4	7
5	6

a. Is f one-to-one?
   No, because to be a one-to-one every input must have an unique output. In other words each of the input cannot have overlapping outputs. For f, f(1), f(3), and f(5) = 6. Also for f(2) and f(4) = 7. Each input have overlapping outputs or relates to the same output.
b. Is f onto?
   No, since an onto function needs to satisfy the condition of for each and every memeber of y∈Y there must exist a matching member x∈X. But f(1), f(3), and f(5) = 6. Also f(2) and f(4) = 7. Y's 8, 9, and 10 have no X elements bounded to them.
c. Is f a correspondence?
   No, because a corresponding function is one that must both an one-to-one function and an onto function. The function f in this case were not either of the two, thus f cannot be a correspondence function.

d. Is g a one-to-one?
   Yes, because each element in X has their own unique output in Y.

e. Is g onto?
   Yes, because each element in Y is bounded to an element in X. There are no Y element that are not left out unlike function f which left out 8,9 and 10.

f. Is g a correspondence?
   Yes, because g is both an one-to-one and an onto function. Therefore it is also a correspondence function.

4.7 Let B be the set of all infinite sequences over {0,1}. Show that B is uncountable using a proof by diagonalization.

For starters, assume B is countable and there exists a correspondence $f : N \to B$. We now create x in B in a manner where it it will not be paired with anything in N. Then consider $x = x_1, x_2, ...$ Then let $x_i = 0$ for the cases where $f(i)_i = 1, and x_i = 1$. if $f(i)_i = 0$ given the fact that $f(i)_i$ is the ith bit of $f(i)$. Thus, this confirming the fact that x is not f(i) for all i since it would differ from f(i) in the ith symbol. As a result, this will cause a contradictiong and proving that B is uncountable.

4.8 Let T = {(i, j, k)| i, j, k ∈ N}. Show that T is countable.
To determine if T is countable we check if it is a one-to-one function. For example, consider the case f(i,j,k) = $1^i, 3^j, 5^k$. This is a one-to-one because if $a \neq b$, then $f(a) \neq f(b)$. Thus, making T countable since it is a one-to-one.

5.1 Show that EQ_{CFG} is undecidable.
Consider the contradiction where EQ_{CFG} is decidable. First, create a decider M for $ALL_{CFG} = \{\langle G\rangle$ where G is a CFG and $L(G) = \Sigma*$.
M = Input $\langle G\rangle$.

1. Make a CFG H with the condition $L(H) = \Sigma*$.
2. Execute decider for EQ_{CFG} for inputs $\langle G, H\rangle$.
3. If the decider accepts then accept, otherwise reject.

M will determine ALL_{CFG} with the assumption that the decider exist for EQ_{CFG}. Due to ALL_{CFG} being undecidable, there is a contradiction. Hence, EQ_{CFG} is undecidable.

5.2 Show that EQ_{CFG} is co-Turing-recognizable.

To do so, consider the Turing Machine, A that recognizes the complement of EQ_{CFG}
A = Input $\langle G, H\rangle$.

1. Generate the strings $x \in \Sigma*$ alphabetically.
2. Consider the cases for each string x
3. Check if $x \in L(G)$ and if $x \in L(H)$ by using A_{CFG}'s algorithm.
4. If one of the cases accepts and the other rejects then accept. Anything else just continue running.

5.3 Find a match in the following instance of the Post Correspondence Problem

$$\left\{\left[\frac{ab}{abab}, \frac{b}{a}, \frac{aba}{b}, \frac{aa}{a}\right]\right\}$$

A possible match:

$$\left[\frac{ab}{abab}, \frac{ab}{abab}, \frac{aba}{b}, \frac{b}{a}, \frac{b}{a}, \frac{aa}{a}, \frac{aa}{a}\right]$$

5.4 If $A \leq_m B$ and B is a regular language, does that imply that A is a regular language? Why or why not?

No, the given condition fails to imply that A is regular. Take the example $\{a^n b^n c^n | n \geq 0\} \leq_m \{a^n b^n | n \geq 0\}$. The reduction of this will first test if the input provided is a member of $\{a^n b^n c^n | n \geq 0\}$. If it is then it will ouput the string ab. Otherwise, it will output the string a.