HW Solution

CS/ECE 374: Algorithms & Models of Computation, Spring 2019 Version: **1.0**

Submitted by:

- ≪**Alan Lee**≫: ≪**alanlee2**≫
- ≪**Joshua Burke**≫: ≪**joshuab3**≫
- ≪**Gerald Kozel**≫: ≪**gjkozel2**≫

(100 PTS.) Prefix it.

Let $L \subseteq \{0, 1\}^*$ be a language defined as follows:

(i) $\varepsilon \in L$.

(ii) For all $w \in L$ we have $0w1 \in L$.

(iii) For all $x, y \in L$ we have $xy \in L$.

And these are all the strings that are in L. Prove, by induction, that for any $w \in L$, and any prefix u of w, we have that $\#_0(u) \geq \#_1(u)$. Here $\#_0(u)$ is the number of 0 appearing in u ($\#_1(u)$ is defined similarly). You can use without proof that $\#_0(xy) = \#_0(x) + \#_0(y)$, for any strings x, y.

2 Solution:

Question: Prove, by induction, that for any $w \in L$, and any prefix u of w, we have that $\#_0(u) \geq \#_1(u)$

1. We will prove by induction on |w|.

2. Base case: |w| = 0 therefore w = u = ε where u is the prefix of w therefore $\#_0(u) = \#_1(u) = 0$.

3. Inductive hypothesis: let k $\geq$ 0 be an integer and u be a prefix of w. Assume for all strings w $\in$ L where |w| is even that $\#_0(u) \geq \#_1(u)$ holds for |w| $\geq$ k.

4. Inductive step: we will prove that all strings w $\in$ L that $\#_0(u) \geq \#_1(u)$ holds for |w| = k+2. We use the increment of k+2 as we will assert that all strings w within L will have an even length.

5. To use the case |w| = k + 2, we will first prove that for all strings w $\in$ L, that |w| is even.
   (i) We will prove this by induction on $|w|$.
   (ii) Base case: $|w| = 0$, $w = \varepsilon$. 0 is even, therefore $|w|$ is even.
   (iii) Inductive hypothesis: Let $k \geq 0$ be an integer. Assume for all $w \in L$ that $|w|$ is even for $|w| \leq k$.
   (iv) We will prove $|w|$ is even for all cases where $w \in L$ and $|w| = k + 1$. From our assumption, when $|w| \leq k$, $|w|$ is even. We have only two ways to generate a string $s \in L$ from w where $|s| > k$:
      - From definition (ii) of L, we can increment w by concatenating it as $0w1$. This increments $|w|$ by 2. Adding 2 to an even number will always result an even number, therefore applying definition (ii) can only produce even-length strings.
      - From definition (iii) of L, we can increment w by concatenating it with another string in L. Under our assumption, |w| is even and any other string s will also have $|v| \leq k$ as even. Concatenating these two strings produces wv, the length of which is the sum of two even numbers. The sum of two even numbers is even, therefore any string $s \in L$ where $|s| > k$ must have even length.

6. Now that we know that |w| is even, there exists only two cases that a string w can be formed such that it is still within the language L.

7. The first case being that the string results from application of definition (ii) where we concatenate a 0 to the front and a 1 to the end of an existing string w. This can be distilled into $\#_0(0w1) = 1 + \#_0(w)$ and $\#_1(0w1) = 1 + \#_1(w)$ from definition ii of L and definition of concatenation.

   Using the assumption in our inductive hypothesis, we have $\#_0(u) \geq \#_1(u)$. Building off of this with our inductive step, we get 2 possible scenarios of the prefix u where u is a sub-string of w or u is the entire string of w.

   In the first scenario where u is a sub-string of w and |u| < |w|, we get $1 + \#_0(u) \geq \#_1(u)$. By definition of addition,our inductive hypothesis holds true for all inductive steps that fall within this case.

   In the second scenario where prefix u is the entire string |u| = |w|, we get $1 + \#_0(u) \geq 1\#_1(u)$. Again by definition of addition, our inductive hypothesis holds true for all inductive steps that fall within this case.

8. The second case is that a string results from the application of definition (iii) where two existing strings x and y in language L are concatenated. From our assumption we can say: $\#_0(ux) \geq \#_1(ux)$, and $\#_0(uy) \geq \#_1(uy)$. A prefix of $x \cdot y$ can be expressed as either a prefix of x, x entirely, x plus a prefix of y, or x plus y entirely. From the given property of string concatenation (use without proof), and since our assumption holds for x and y individually, then this assumption must hold for $x \cdot y$. Therefore all iterations on w for our inductive hypothesis hold true.