Assignment 2 – FIT3155

Name – Sadeeptha Bandara

ID – 30769140

Question 3: Prove Elias Omega code is prefix-free

Proposition

The Elias Omega code E_x, for any integer $x \in \mathbb{Z}^+ = \{1, 2, 3, \dots \infty\}$ cannot be a prefix of the Elias Omega code E_y for any integer $y \in \mathbb{Z}^+ = \{1, 2, 3, \dots \infty\}$ where $x \neq y$.

Definitions

Consider the Elias codeword E_x for any integer $x \in \mathbb{Z}^+$

$$E_x = L_n \, L_{n-1} \dots L_2 L_1 C_x$$

where L_j $(1 \leq j \leq n)$ represents a length component of the Elias codeword

$$L_j = \begin{cases} FlipFirst(Bitlen(C_x) - 1) & j = 1 \\ FlipFirst(Bitlen(L_{j-1}) - 1) & 2 \leq j \leq n \end{cases}$$

Equation 1

and n is such that $L_n = 0$. FlipFirst refers to a function that flips the first bit of a provided integer to 0. Thus, note that L_j contains one or more padded zeros. (Eg: 0010 is not shortened to 10)

C_x code component represents the minimum binary code of x. The most significant bit of the code component must therefore be 1.

The $L_n \, L_{n-1} \dots L_2 L_1$ component is collectively referred to as the Elias length component and C_x is referred to as the Elias code component

Lemma

A length component L_j $(1 \leq j \leq n)$ is unique to the bitlength of its preceding length component/ code component.

Lemma 1

This can be reasoned intuitively as follows. Consider a length component/code component x with a bit length k. The next length component L_j is given by

$$L_j = FlipFirst(k-1)$$

Different values of k will yield different L_j since each different value is subjected to the same subtraction and bit flip operation if the different k is of the same bit length. If k is of different bit lengths, then L_j cannot be the same since k is of different length.

Therefore, the first length component L_1 is unique to the bit length of C_x. **Similarly, this logic can be extended to conclude that the Elias length component for a particular Elias code component $\boldsymbol{C_x}$ is unique to the bit length of $\boldsymbol{C_x}$**

$$k_{C_x} = Bitlen(C_x)$$

$$L_1 = f\left(k_{C_x}\right)$$

$$L_2 = f\left(k_{L_1}\right)$$

$$\dots$$

$$L_j = f\left(k_{L_{j-1}}\right)$$

f is a one-to-one function. Since the length component is made up of a chain of one-to-one functions, it follows that,

$$Length\ component = g(k_{C_x})$$

Where g is a one-to-one function. Therefore, the Elias length component for a particular Elias code component C_x is unique to the bitlength of C_x

Proof by contradiction

Assume an Elias codeword E_x for an integer $x \in \mathbb{Z}^+$, is a prefix of an Elias codeword E_y for an integer $y \in \mathbb{Z}^+$, where $x \neq y$

$$E_y = L_{n_1}^y\ L_{n_1-1}^y \ \dots\dots\ L_2^y\ L_1^y\ C_y = \ L_y\, C_y$$

$$E_x = L_{n_2}^x\ L_{n_1-1}^x \dots\dots\ L_2^x\ L_1^x\ C_x\ = L_x\, C_x$$

where $L_j^k,\ k\ \in \{x, y\}$ is given by Equation 1 as defined previously. $L_k,\ k\ \in \{x, y\}$ represents the Elias length component for the two integers.

Consider the cases where the length components of E_x and E_y are equal or inequal.

Case 1: $L_x = L_y$

$$L_x = L_y$$

As per Lemma 1, if the Length components are equal, the bit lengths of C_x and C_y should be the equal as well

$$Bitlen(C_x) = Bitlen(C_y) \to (1) \quad (Lemma\ 1)$$

As per our initial assumption that the Elias codeword E_x is a prefix of the Elias codeword E_y, and since $L_x = L_y$,

$$C_x \text{ should be a prefix of } C_y$$

There should exist a k such that

$$C_y = C_x\, k$$

where k is the remaining substring after C_x within C_y. Therefore,

$$Bitlen(C_y) = Bitlen(C_x) + Bitlen(k)$$

However, as per (1)

$$Bitlen(C_y) = Bitlen(C_x)$$

$$\therefore\ Bitlen(k) = 0$$

Therefore, since the length of k is 0, C_y should be equal to C_x

$$C_y = C_x$$

However, as per (1), C_y and C_x have the same bitlength. Minimum binary codes for two different integers of the same bitlength must be unique. Therefore, x must equal y

$$x = y$$

However as per our initial assumption $x \neq y$.

Therefore, for all cases where $L_x = L_y$, E_x cannot be a prefix of E_y as per proof by contradiction.

Case 2: $L_x \neq L_y$

Consider a region k as follows

$$E_y = E_x\ k$$

where k is the remaining substring after E_x within E_y.

$$E_y = \underbrace{L_n^x\ L_{n-1}^x \ldots\ldots L_2^x\ L_1^x\ C_x}_{E_x}\ k$$

Consider the first length component of E_x. As per the definition of L_j^x, $(1 \leq j \leq n)$.

$$L_n^x = 0$$

Similarly,

$$L_n^y = 0$$

The first components of both E_x and E_y must be the same. Consider the next component of both Elias Omega codewords, $Comp_x$ and $Comp_y$ respectively.

As per Lemma 1, a particular length component is unique to the bit length of its previous component. Therefore, it must be true that,

$$Bitlen(Comp_x) = Bitlen(Comp_y)$$

Since, E_x is a prefix of E_y, it must hold true that,

$$Comp_x = Comp_y$$

As per this logic, all successive components from the start of E_x must have a matching component in E_y. $\rightarrow$ (1)

Since $L_x \neq L_y$, L_x must be either longer or shorter than L_y. However, since E_x is a prefix of E_y, and since the number of bits in each component is monotonically increasing with each new component,

$$Bitlen(L_x) < Bitlen(L_y)$$

As each component within E_x has a matching component in E_y and since $Bitlen(L_x) < Bitlen(L_y)$, the entirety of E_x must be contained within L_y

$$L_y = E_x\, k$$

where k is the remaining substring after E_x within L_y. Since each component within L_y matches with a component within E_x, the code component of E_x must match a length component of E_y.

There must exist a L_j^y within L_y such that

$$L_j^y = C_x$$

However, a length component L_j^y necessarily starts with a padded zero in front and a minimum binary code C_x necessarily starts with a 1 as the most significant bit. Therefore, it is not possible that $L_j^y = C_x$.

$$L_j^y \neq C_x$$

As such, for all $L_x \neq L_y$, E_x cannot be a prefix of E_y as per proof by contradiction.

In conclusion, the Elias Omega code $\boldsymbol{E_x}$, for any integer $\boldsymbol{x} \in \mathbb{Z}^+ = \{\mathbf{1}, \mathbf{2}, \mathbf{3}, \dots\ \infty\}$ cannot be a prefix of the Elias Omega code $\boldsymbol{E_y}$ for any integer $\boldsymbol{y} \in \mathbb{Z}^+ = \{\mathbf{1}, \mathbf{2}, \mathbf{3}, \dots\ \infty\}$ where $\boldsymbol{x} \neq \boldsymbol{y}$.