15-455: UCT	**K. Sutner**
Assignment 8	**CHANGED** Due: **April 11**, 2022.

1. Infinity Encodings (20)

Background

In class we showed how to define a prefix encoding for binary strings that adds essentially just a log factor to the length of the string. The version given in class is a bit clumsy, here is a slightly better approach.

To avoid pesky edge cases, we only consider words x of length at least 2. Recall that $\mathsf{len}^*(x)$ is the least $k \geq 1$ such that $\mathsf{len}_k(x)$ has 2 digits. More precisely, think of $\mathsf{len}, \mathsf{len}_i : \mathbf{2}^\star \to \mathbf{2}^\star$ and $\mathsf{len}^* : \mathbf{2}^\star \to \mathbb{N}$. The binary string $\mathsf{len}(x)$ indicating $|x|$ has its MSD on the left, there are no leading zeros. So $\mathsf{len}^*(ab) = \mathsf{len}^*(abc) = 1$ but $\mathsf{len}^*(x) \geq 2$ for longer strings.

We keep the basic prefix coding functions. Here coding function means that a string function is injective and has an easily decidable range, the set of all code words; the inverse decoding function on those code words must also be easily computable. Prefix means that the set of code words is a prefix language.

$$\begin{aligned} E(x) &= x_1 0 x_2 0 \ldots x_n 1 \\ E_0(x) &= E(x) \\ E_{i+1}(x) &= E_i(\mathsf{len}(x))\, x \end{aligned}$$

Here are two "infinity" versions, in both cases let $k = \mathsf{len}^*(x)$:

$$\begin{aligned} E^\infty(x) &= E_k(x) \\ E_\infty(x) &= \mathsf{len}_k(x)\, 0\, \mathsf{len}_{k-1}(x)\, 0 \, \ldots \, |x|\, 0\, x\, 1 \end{aligned}$$

as opposed to the old $E(k)E_k(x)$ that makes the value of k explicit. So with these encodings, any string x of length 20000 turns into

$$\begin{aligned} E^\infty(x) &= E_4(x) = 1011 \;\; 100 \;\; 1111 \;\; 100111000100000 \;\; x \\ E_\infty(x) &= 11 \; 0 \; 100 \; 0 \; 1111 \; 0 \; 100111000100000 \; 0 \; x \;\; 1 \end{aligned}$$

where the extra spaces are added for visually clarity, they are missing in the actual code.

Task

A. Show that the basic functions E_i really are prefix encodings.

B. Show that E^∞ is an encoding.

C. Show that E_∞ is a prefix encoding.

Comment You probably want to establish a few simple facts about the sequence $\mathsf{len}_i(x)$.

2. Kolmogorov versus Palindromes (30)

Background

Suppose M is a one-tape Turing machine recognizing palindromes over $\{0,1\}$. We say that M crosses tape cell number i if either

- the head moves right from i to $i+1$, or
- the head moves left from $i+1$ to i.

We can construct of a crossing sequence $((p_1,s_1),(p_2,s_2),\ldots)$ of all crossings of position i keeping track of the state p_i and the read symbol s_i at the moment of crossing (before the move). Note that right/left crossings must alternate.

Write $T(x)$ for the running time of M on input x, and assume that the machine always halts with the head on the right end of the string (it starts on the left). To streamline the argument a bit, it's best to consider input of the form $x = z0^n z^{\mathsf{op}}$ where $|x| = n$. The region $[n+1, n+2, \ldots, 2n]$ is called the desert. Note that every position in the desert has at least one crossing.

Task

A. Show that some position I in the desert must have a crossing sequence of length $m \leq T(x)/n$.

B. Show that z is the unique string of length n such that input $z0^{I-n}$ produces this crossing sequence.

C. Exploit part (B) to give a compact description of x and conclude that we cannot have $T(x) = o(n^2)$.

3. Kolmogorov versus Primes (30)

Background

One can (ab)use Kolmogorov-Chaitin complexity to show that there are infinitely many primes, though many would argue that the original argument is far superior. But, with a little bit of extra effort, one can push this argument to get a fairly good estimate for the density of primes. Write $\pi(n)$ for the number of primes up to n. The celebrated and difficult prime number theorem says that $\pi(n) \approx n/\log n$. We will settle for a weaker claim: $\pi(n) \geq cn/\log^2 n$

Write $p_1, p_2, \dots$ for the sequence of primes, so that for any number n we have a unique decomposition $n = \prod_{i \leq m} p_i^{e_i}$, $e_i \geq 0$.

Task

A. Use Kolmogorov-Chaitin complexity to show that there are infinitely many primes.

B. Use Kolmogorov-Chaitin complexity to prove $\pi(n) \geq cn/\log^2 n$, for some constant c and infinitely many n.

Comment For the last part, use the fact that a number n can be decomposed into its largest prime factor p and n/p; the prefix coding functions E_k also come in handy.

4. Uninspired Sets (20)

Background

Let $K(x \mid y)$ be the conditional Kolmogorov-Chaitin complexity of $x \in \mathbf{2}^\star$, given y. For any set $A \subseteq \mathbb{N}$ write $A_n = A \cap \{0, 1, \ldots, n-1\}$ for the initial segment of A of length n. Think of A_n as bitvector of length n.

As we have seen, incompressibility with respect to Kolmogorov-Chaitin complexity is akin to randomness: there are no particular patterns one could exploit to obtain a shorter definition. How about the opposite notion? Call $A \subseteq \mathbb{N}$ uninspired if there is a constant c such that

$$K(A_n \mid n) \leq \log n + c.$$

So only some $\log n$ bits are needed to describe the corresponding bitvector of length n, given n.

Task

A. Show that any decidable set A is uninspired.

B. How about the Halting Set H? State whether H is uninspired and explain your reasoning.

C. Repeat for the complement of the Halting Set.