Homework 8

Joe Baker, Brett Schreiber, Brian Knotten

January 31, 2018

11

- Show that the set of semi-incompressible strings is not computable.

  Assume that the set of semi-incompressible strings is computable to reach a contradiction.
  Therefore $\exists$ TM M that computes the set of semi-incompressible strings.
  Consider the TM N that for each string x of size n:

  Run M. If x is generated by M print x and halt.

  N has n hard-coded into it using $log_2(n)$ bits to store n and the rest of N's encoding is represented using some constant amount of bits k. Therefore $|N| = O(log_2 n)$, which is $< \sqrt{n}$ for a sufficiently large n, i.e. $\exists$ a TM N that can represent x where $|N| < \sqrt{|x|}$, contradicting the semi-incompressibility of x.

- Show that there are finitely many incompressible strings with an equal number of 1's and 0's.

  Every string s where s has the same number of 0s and 1s and $|s| \geq 4$ is compressible to size $|s-1|$.
  For every string s of length n, where $n \geq 4$, there are enough strings of length $n-1$ to hold all strings in s.
  The number of strings with the same number of 0s and 1s make up $\frac{1}{\sqrt{(n)}}$ of all strings of length n. Consider flipping a coin n times and recording a 0 if it lands on heads, 1 if it lands on tails. This has a probability of happening $\frac{1}{\sqrt{(n)}}$ times. So there are $\frac{2^n}{\sqrt{n}}$ strings of length n that have the same number of 0s and 1s.

  So if $\frac{2^n}{\sqrt{n}} \leq 2^{n-1}$, then there are enough strings of length $n-1$ to hold each string of length n which have the same number of 0s and 1s.

  $$\frac{2^n}{\sqrt{n}} \leq 2^{n-|M|} \Leftrightarrow n \geq 2^{2|M|}$$
  $$\frac{2^n}{2^{n-|M|}} \leq \sqrt{n} \Leftrightarrow n \geq 2^{2|M|}$$
  $$2^{|M|} \leq \sqrt{n} \Leftrightarrow n \geq 2^{2|M|}$$
  $$2^{2|M|} \leq n \Leftrightarrow n \geq 2^{2|M|}$$

- Show that the set of incompressible strings contains no infinite subset that is recursively enumerable.

  Let I be the set of incompressible strings and let $S \subset I$ be an infinite subset of I. Assume that there is an infinite subset of incompressible strings S that is recursively enumerable. Then there must be a TM M which can enumerate S. There are two cases:

  1. If $\exists x \in S$ where $|x| > |M|$, then x is compressible, which is a contradiction
  2. If $!\exists x \in S$ where $|x| > |M|$, then S is finite, which is also a contradiction

  In any case, the existence of both M and S creates a contradiction so our assumption is false and there is no infinite subset of incompressible strings that is recursively enumerable.

- Show that the set of compressible strings is recursively enumerable.

  A string is compressible if it is not incompressible.

The set of all strings is enumerable. Consider each string to be an integer of base $|\Gamma|$ where $|\Gamma|$ is the size of the alphabet.
For an alphabet $\Gamma = \{0, 1\}$, then $ALL\_STRINGS = \{\epsilon, 0, 1, 00, 01, 10, 11, 000, 001, ...\}$ Consider the TM M which:
For each string s in $ALL\_STRINGS$:
    for each string t in $ALL\_STRINGS$ where $|t| < |s|$
        if t treated as a Turing machine, describes s, output s and break to outer for loop
Since $|t|$ is less than $|s|$, and t perfectly describes s, then t is a compressed version of s. Therefore, s is compressible. Since M checks all strings which are less than the given string s, then M will eventually find a compressed version of s if it exists. Therefore $L(M)$, the set of compressible strings, is recursively enumerable.

12

Show that for any $c > 0$, there exists strings x and y such that $K(xy) > K(x) + K(y) + c$, where $K(x)$ is the Kolmogorov complexity of x.

Proof by contradiction:
Assume that $\exists c\ \forall x, y$ such that $K(xy) \leq K(x) + K(y) + c$. This statement suggests that any two strings can be represented more compactly when concatenated then compressed than when compressed individually and then concatenated. However there is a contradiction. $K(xy)$ must have a Turing machine M which is the most compact machine that can produce xy for any pair of strings, and our assumption states that $|M| \leq K(x)+K(y)+c$. But if M can produce xy for all strings x and y, then it needs to know where to split the two strings. Storing the information on where to split the two strings requires more than a constant value since the size of either string can vary. Thus, we have a contradiction because $K(xy) \leq K(x)+K(y)+c$ is false for all string pairs. Finally, $\forall c > 0\ \exists x, y$ such that $K(xy) > K(x) + K(y) + c$.