Please indicate how much time you spent on this assignment.

Bonus Problem: Is $K(x)$ really uncomputable?

A skeptical 341 student is unconvinced that $K(x)$ is not computable. In fact, they give you the following proof that $K(x)$ is computable.

**Proof.** Let $N$ be the following Turing machine.

$N = \begin{quote}
\text{On input } x, \\
1. \text{Iterate over all possible strings } \langle M, w \rangle \text{ in lexicographical order until you find the shortest one that computes } x. \\
2. \text{Output the length of } \langle M, w \rangle \end{quote}$

We begin by noting that for every string $x$, there exists some string $\langle M, w \rangle$ where running $M$ on $w$ outputs $x$. Therefore $N$ will always halt, since it is guaranteed to eventually find one such string. Furthermore, since it iterates over the strings in lexicographical order, it will always find the smallest string $\langle M, w \rangle$ that outputs $x$. Therefore $N$ computes $K(x)$.

(a) Identify the error in the above argument.

(b) Articulate in your own words why $K(x)$ is uncomputable. (Even more bonus points if you give a proof!)

Bonus Problem: We want more information!

Suppose we want to write an algorithm that takes a string as input and outputs a string that has more descriptive information than its input. Prove that if $f : \Sigma^* \rightarrow \Sigma^*$ is a computable function, then there exists a constant $c$ such that

$$K(f(x)) \leq K(x) + c$$

for all strings $x \in \Sigma^*$.

(Note that an immediately corollary of this fact is that it is impossible to write an algorithm that adds more than a constant amount of descriptive information to its input.)