

Week 9: Is “Heterological” Heterological?

Response by: Benjamin (aqn9yv, dl2ru, ht6xd, iad4de, jmn4fms, lw7jz)

Problem 5 *Accepts in k steps*

Consider the Language:

$$A_k = \{M \mid M \text{ is the description of a no-input Turing Machine which accepts in } k \text{ or fewer steps} \}.$$

Show that A_k is computable for every choice of $k \in \mathbb{N}$.

Proof.

We prove the statement by proving that it is possible to construct a Turing Machine M for any k , such that M computes A_k for that k .

We construct the machine in two parts, the simulation machine and the counting machine. The two machines are each DFAs (with a shared tape) and collectively constitutes a NFA, which could be converted into an even larger DFA. The simulation machine simulates the ACCEPTS function, and the counting machine functions as a counter program that counts the number of steps the simulation machine have been executing. If the counting machine counts to k while the simulation machine has not finished its execution, the counting machine will break the execution and return false.

The construction of the simulation machine is assumed to be available from lecture.

We construct the counting machine using k states, each represent a step taken on the simulating machine. The counting machine take any character in the alphabet as input to transition from the i^{th} state to the $(i + 1)^{th}$ state for any $i < k$. If the counting machine reaches its final state, and the simulation machine hasn't provided an output yet, then the counting machine will halt the entire machine and return false. If the simulating machine returns a value before k steps, the combined machine will take the output of the simulating machine as the output of itself.

Thus we have constructed a machine that computes the given language, also provided that it will halt in k steps.

□