For with what judgment ye judge, ye shall be judged: and with what measure ye mete, it shall be measured to you again.

Matthew 7:2

Proof of proposition: A human H ought not to be computing an arbitrary human H'.

Assumption. *The Church-Turing Thesis is true: everything that is physically computable is computable by some Turing machine.*

Definition 1. *A human H is a thing that does computation and is in the physical world.*

Remark: In other words, a human H is an automaton to which the Church-Turing Thesis applies; for each thought process of human H, there exists a Turing machine.

Definition 2. *A human H is free if and only if H is uncomputable.*

Corollary: A human H is not free if and only if H is computable.

Definition 3. *We say a human H "ought not to be" executing some Turing machine M in the case that H is not free if H is executing M.*

Proposition 1. *A human H is at most Turing-complete.*

Proof: This follows from Assumption and Definition 1.

Proposition 2. *There exists no Turing machine M that computes the output of an arbitrary Turing machine A.*

Proof: This follows from the undecidability of the halting problem.

Definition 4. *An automaton S is said to be "stronger" than an automaton W if and only if the functions W can compute is a strict subset of the functions M can compute. Conversely, W is "weaker" than S if and only if S is said to be "stronger" than W.*

Remark: This definition exists purely for the sake of linguistic convenience. In each subsequent proposition, replace "stronger" or "weaker" with the formal definition here.

Proposition 3. *For some automaton M, if M is computing the output of an arbitrary Turing machine A, M is either stronger than or weaker than a universal Turing machine.*

Proof: By Proposition 2, no Turing machine M computes the output of an arbitrary Turing machine A. Therefore, if M computes the output of an arbitrary Turing machine, M is not a Turing machine. In particular, M is not a universal Turing machine. There are two possibilities for M. (1) M is Turing-complete and has extra computing capabilities. For example, M may be a universal Turing machine with a halting problem oracle. (2) M is sub-Turing-complete, that is, there are Turing machines which M cannot simulate. Therefore, in this case, M is either stronger than or weaker than a universal Turing machine.

Proposition 4. *If a human H is computing the output of an arbitrary Turing machine A, H is weaker than a universal Turing machine.*

Proof: By Definition 1, a human H is an automaton. By Proposition 4, if an automaton H computes the output of an arbitrary Turing machine A, H is either stronger or weaker than a universal Turing machine. By Assumption, a human H is no stronger than a Turing-complete machine. Therefore, H is weaker than a universal Turing machine.

Proposition 5. *If a human H is computing the output of an arbitrary Turing machine A, H is computable by some Turing machine.*

Proof: By Proposition 5, if a human H is computing the output of an arbitrary Turing machine A, H is weaker than a universal Turing machine. Therefore, H is a sub-Turing-complete machine.

Lemma 1: There exists a Turing machine that can compute the outcome of any sub-Turing-complete machine. *Proof is left as an exercise for the reader.*

By Lemma 1, if H is a sub-Turing complete machine, H is computable by some Turing machine.

Proposition 6. *If a human H is computing the output of an arbitrary Turing machine A, H is not free.*

Proof: By Proposition 6, if a human H is computing the output of an arbitrary Turing machine A, H is computable by some Turing machine. By the corollary to Definition 2, if H is computable, H is not free.

Proposition 7. *If a human H is computing a free human H', H is not free.*

Proof: By Definition 2, a human H' is free if and only if H' is uncomputable. By Proposition 2, H' is at most Turing-complete. Because H' is uncomputable, H' must be at least Turing-complete. Therefore, H' is exactly Turing-complete. To compute the output of a Turing-complete machine is tantamount to computing the output of an arbitrary Turing machine. By Proposition 7, if a human H computes the output of an arbitrary Turing machine A, H is not free. Therefore, if a human H computes the output of a Turing-complete machine H', H is not free. Therefore, if a human H computes a free human H', H is not free.

So far, so good. However, at this point, one problem remains. A human H may compute some H' and simply claim that H' is not free, therefore H is free. But how should H know if H' is free or not? We show that there is no Turing machine to do just that. This lets us squeeze out a stronger result: *If a human H is computing an arbitrary human H', H is not free.*

Proposition 8. *There is no Turing machine M that takes as input an arbitrary human H and outputs whether H is free or not.*

Proof: Suppose such a Turing machine M exists. Then M takes as input an automaton H and outputs whether H is an arbitrary Turing machine or not. If H were an arbitrary Turing machine, M could not know if H halts or not. If H were a sub-Turing-complete machine, then M can run H until it halts. Any Turing machine that halts can be simulated by a sub-Turing-complete machine. If H were to halt, H can be simulated by a sub-Turing-complete machine. Therefore M is equivalent to the solution to the halting problem. Therefore no M exists.

Remark: Clearly, there exists a Turing machine M that takes as input a human H with a specific semantic description – namely, that H is computing a free human H' – and outputs whether H is free or not: that Turing machine is described by Propositions 1-8. We may gain such a semantic description about H through, for example, something H has said or done. However, we are talking here about an *arbitrary* human H that may or may not possess this semantic description. We have shown that, in this general case, there exists no such M.

Proposition 9. *If a human H is computing an arbitrary human H', H is not free.*

Proof: By Proposition 9, no Turing machine M exists that takes as input an arbitrary human H and outputs whether H is free or not. The rest of the proof mirrors the structure of the proof to Proposition 8.

Proposition 10. *A human H ought not to be computing an arbitrary human H'.*

Proof: This follows from Definition 3, Proposition 10 and Assumption.