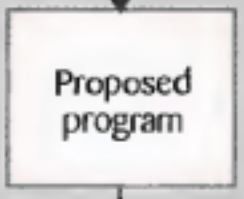
First: Propose the existence of a program that,

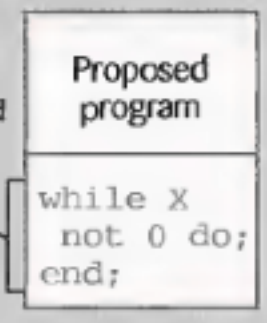
given any encoded version of a program



will halt with variable X equal to 1 if the input represents a self-terminating program, or with X equal to 0 otherwise.

Then: If such a program exists, we could modify it by

adding a while-end structure

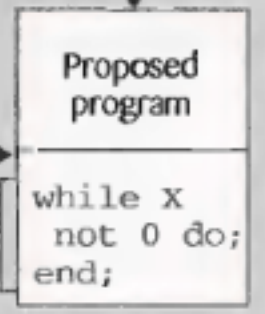


to produce a new program

Now: If this new program were self-terminating and

we started it with its own encoding as its input

execution would reach this point with X equal to 1,



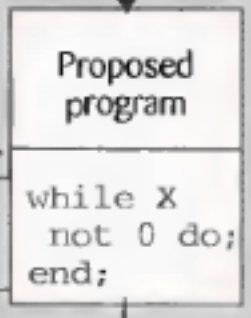
so execution would become trapped in this loop forever;

i.e., if the new program is self-terminating, then it is not self-terminating.

However: If this new program were not self-terminating and

we started it with its own encoding as its input,

execution would reach this point with X equal to 0,



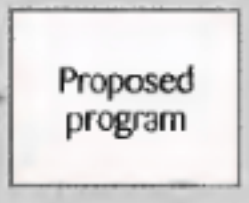
so this loop would be skipped

and execution would halt;

i.e., if the new program is not self-terminating, then it is self-terminating

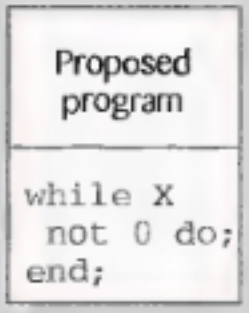
Consequently:

The existence of the proposed program



would lead to

the existence of a new program



that is neither self-terminating nor not self-terminating

so the existence of the proposed program is impossible.