Where, and how, does code's syntax start becoming, or is it forbidden to become, and intentional act of artistic expression (semantic?) for the computer?

systems that cannot realize that won't terminate

Godel, Escher, Bach

"It is an inherent property of intelligence that it can jump out of a task which it is performing, and survey what it has done." (37)

Machines can be made to be totally unobservant; and people cannot. (37)

Halting Problem

Machines cannot realize that won't terminate
definition of creativity according to Law in Patent disputes

@makomarino

@jeremydouglas
cultural norm

addressing meaning in programming languages

@gtorre

Halting Problem

Definition of creativity according to Law in Patent disputes

@makomarino

@jeremydouglas
cultural norm

"What is this program's output?" but "What does it mean?" Or rather how does its shape determine its meaning. What in the original language allows "AI Prison" to mean what it does? Could this other language express the same idea? How does one implementation capture the spirit of the program more than another, the intention as explained by the artist, or the meaning that we assign to it? What affordances would I have to tap into for the same effect? (And of course, it could never be the exactly the same.)

Speculating on how AI could break free from the prison:

1. Cyclic redundancy check (CRC) for a number
2. DMA attack from other AI

Critical Code Studies Working Group 2018