

Code and computer's creativity? hmmm...

```

1. //AI PRISON by Giuseppe Torre
2. #include<iostream>
3. int main()
4. {
5. int i;
6. while(i ==0){
7.     std::cout << i ;
8. }
9.     std::cout <<"RUN!"<< std::endl;
10. return0;
11. }

```

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

Critical Code Studies Working Group 2018

computer creativity as:

canon war

draw similarities with canon wars of the 80s' and 90's. Higher education "war" on which author needed to be taught to achieve a multiculturalist/balanced curriculum

@melstanfill

interesting/wild take. maybe a blend of postmodernism and positivism has played a greater role in this rhizomic/fair way of thinking about creativity

creativity needs redefinition

there is no agreement on what creativity is for human. That is good. No need to agree on this in relation to computers

@gtorre

Definition of creativity according to Law in Patent disputes

cultural norm

algorithm complexity is irrelevant. In relation to machines there is no reason to believe that more syntax gives rise to semantic (Searle)

@jeremydouglass

@belljo

systems that cannot realize that won't terminate

@makcmarino

Godel, Escher, Bach

Halting Problem

@erwig

the model is in part an alien artifact.

"But this line of thinking raises an odd question in my mind: if two humans with different perspectives can find one another creative, can two algorithms find each other creative? " --> siding on complexity

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

Frege' sense and reference

"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)."

addressing meaning in programming languages

simplicity vs complexity of code

skeptical of Frege. Language always has a material aspect that doesn't cease to matter when it participates in a semantic system. That material seeps into sense, and sense seeps into reference.

@ebuswell

comparison with 10 PRINT porting

porting and the loss of meaning

Speculating on how AI could break free from the prison:

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

10 PRINT looks at the output (forward)

AI PRISON looks back into the code through a sort of "speculative counterhistory" informed by a regress argument critiquing the idea of creativity in machines. C++ code syntax is the base material for this critique(output is secondary).

@gtorre

"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.)

Heideggerian starting point also worth considering: "we do not speaks language; language speaks us."

@gtorre