"What is the answer to this question?"

or,

Self-reference and inherent limitations of computers and the brain

self-ref·er·en·tial *(sĕlf'rĕf'ə-rĕn'shəl) adj.*
Def. See "self-referential."

Chucky Ellison
July 12, 2006

- This sentence is false.
- If the meanings of "true" and "false" were switched, then this sentence wouldn't be false.
- This sentence no verb.
- "is a sentence with no subject" is a sentence with no subject.
- Disobey this command.
- This sentense contains exactly three erors.
- This sentence is not about itself, but about whether it is about itself.
- I don't care who wrote this sentence; whoever he is, he's a damn sexist.
- This sentence is a !!!! premature punctuator
- Only the fool would take trouble to verify that this sentence was composed of ten a's, three b's, four c's, four d's, forty-six e's, sixteen f's, four g's, thirteen h's, fifteen i's, two k's, nine l's, four m's, twenty-five n's, twenty-four o's, five p's, sixteen r's, forty-one s's, thirty-seven t's, ten u's, eight v's, eight w's, four x's, eleven y's, twenty-seven commas, twenty-three apostrophes, seven hyphens, and, last but not least, a single !

Self-Reference

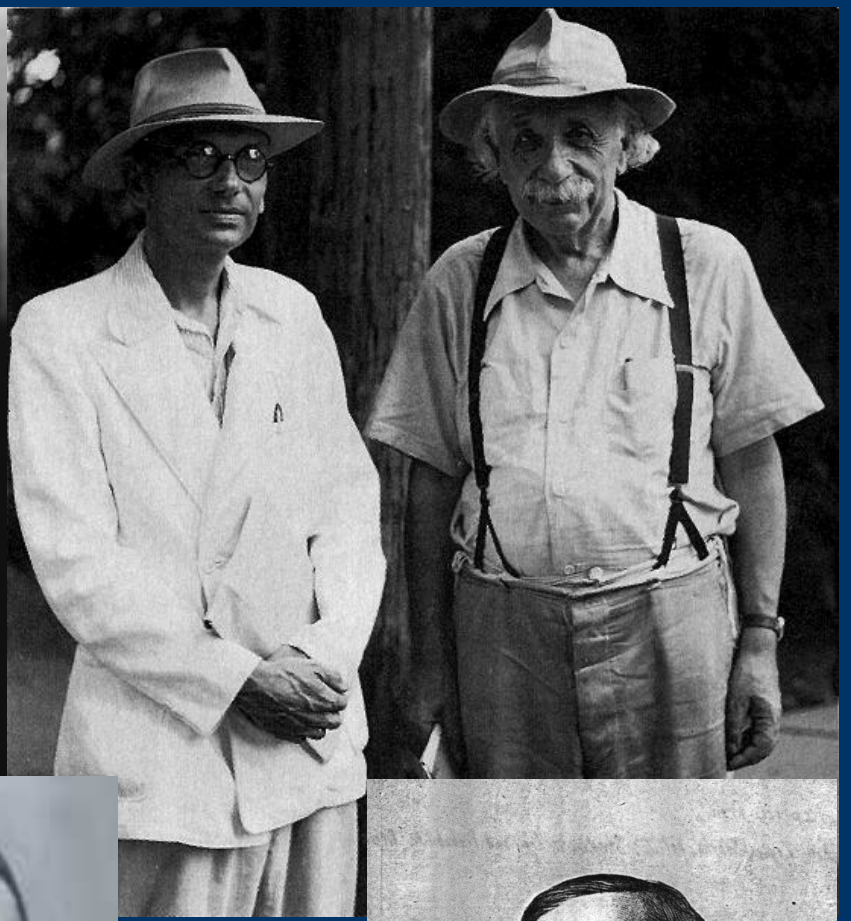
- Degrees in computer science and mathematics from North Carolina State University
- Working on PhD in computer science at Universitiy of Illinois at Urbana-Champaign
- Started playing with computers when I was four
- Thought I was bad at math until sophomore year of high school (now I think I'm mediocre)

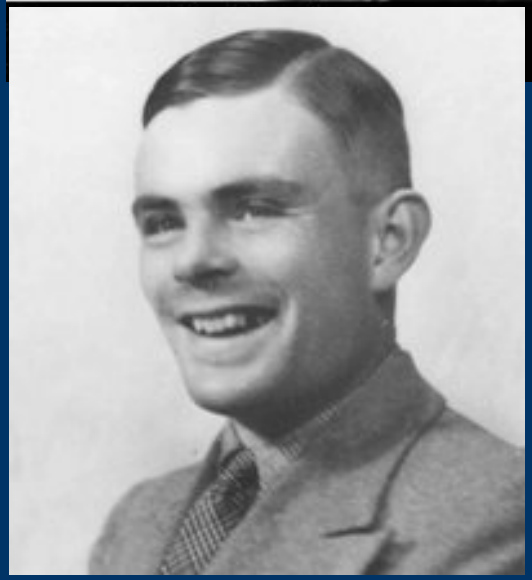
Overview

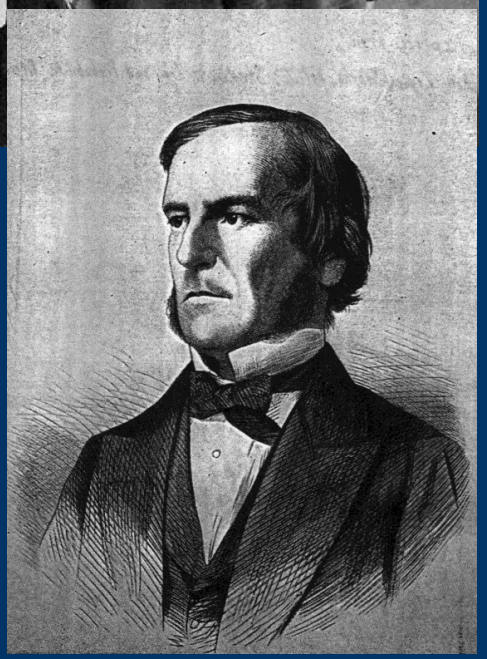
- History
- Gödel's incompleteness theorems
- Formal computing
- Halting problem
- Recursion
- Reflection
- Self-modifying code
- Quines

History, Background

- Rich formal background before engineering
- Formal vs practical is a false dichotomy
  - Computer scientists develop new mathematics
  - Mathematicians develop new algorithms and technology







Halting problem, 1

- Given a description of a program and its initial input, determine whether the program, when executed on this input, ever halts (completes).
- That is to say, your goal is to write the following function

```
function your_goal(string program, string input)
    if halts(program, input)
        return true
    else
        return false
```

Halting problem, 2

```
function trouble(string s)
    if your_goal(s, s)
        loop forever
    else
        return true
```

Recursion

```
function factorial(n) {
  if (n <= 1)
    return 1;
  else
    return n * factorial(n-1);
}
```

```
let rec map f l =
   match l with
     [] -> []
      | hd :: tl -> f hd :: map f tl;;
```

Reflection

```
# without reflection
Foo->new->hello;

# with reflection
my $class = "Foo";
my $method = "hello";
$class->new->$method;
```

```
// Without reflection
Foo foo = new Foo();
foo.hello();

// With reflection
Class cl = Class.forName("Foo");
Method method = cl.getMethod("hello", null);
method.invoke(cl.newInstance(), null);
```

Self-modifying code

- Remove one-time conditional within loop
- Generation of code

Quines

```
| prog quote |
quote := Character value: 39.
prog := '| prog quote |
quote := Character value: 39.
prog := .
Transcript show: (prog copyFrom: 1 to: 53).
Transcript show: quote asString; show: prog; show: quote
  asString.
Transcript show: (prog copyFrom: 54 to: 212).
'.
Transcript show: (prog copyFrom: 1 to: 53).
Transcript show: quote asString; show: prog; show: quote
  asString.
Transcript show: (prog copyFrom: 54 to: 212).
```

"This is not a pipe."