CSCI 301 - Lecture 21

Theory of Computation Intro

Theory of Computation:

Computability Theory	Complexity Theory
"What can be computed?"	"How 'hard' is this problem?"

Halting problem: (Example of computability)

H: Given a program P, and an input I, return true if P terminates (halts) on input I.

```
P
(+ 2 2)
```

```
Q(i)
while i>0 :
    i+=1
```

H(P, I)
- → true
- → false

Thm: H doesn't exist.

Proof: Suppose H exists.

Let Z:

```
Z(string x)
 if H(x, x):
     loop forever
 else:
     return
```

Run: Z(Z)

H(Z,Z) → true

H(Z,Z) → false

P vs NP (Example of Complexity Theory)

$P \subseteq NP$

$P \stackrel{?}{=} NP$ ⟵ verifiable in polynomial time

↑ solvable in polynomial time

Automata Theory

↑ sing. "automaton" - model of a computer

model of "problem" - languages accepted by an automaton

Toll gate: toll 15¢

accepts 5¢, 10¢ coins