Announcements

- **HW10** due Friday (Solving hard problems using integer programming and algorithms with huge numbers used for crypto)

For coding using Python, please do not use the matrix interface by importing numpy and scipy! Our Gradescope autograder does not support these two packages

- Last homework: **HW11** on computability ~~will be~~ is due May 1st

Section plans: week of April 27-28: quiz + computability practice

week of May 4-5: quiz + review of material since the prelim

Final (cumulative): Saturday May 9th at 9am.

Thursday May 7: 4820 review led by a few TAs

- Options for last couple lectures (and other questions you may have): **see survey** (see pointer pinned on Ed)

winning option so far: divide and conquer

Join by Web PollEv.com/evatardos772

How is the coding problem going?

A. I have not started
B. Cool to have access to such a powerful system
C. Done, it was Ok, but annoying
D. Working on it, and struggling to see how to do this

Today: Halting problem not decidable proof technique: diagonalization

Real $[0,1]$ not countable

countable 1, 2, ...

e.g. $\mathbb{N}$, $\mathbb{Z}$ $0, 1, -1, 2, -2, 3, -3$

rationals: $0, 1, \frac{1}{1}, \frac{1}{2}, \ldots$ · see 2800

Proof by contradiction

$x \in [0,1]$ $0.10\ldots$ in binary

$0.00\ldots$

$0.101\ldots$

$[0,1]$

table digits after the leading 0

Idea of diagonalization

Now consider diagonal 00011...

digit i is i^{th} after 0. on the i^{th} number

y = diagonal with entries swapped $0 \leftrightarrow 1$

$y = 0.11100\ldots \in [0,1]$

Where is y in our list?

not on the list: n^{th} digit disagrees with n^{th} on the list

$\Longrightarrow$ contradiction $[0,1]$ reals cannot be listed

Programs, input and 0/1 sequences

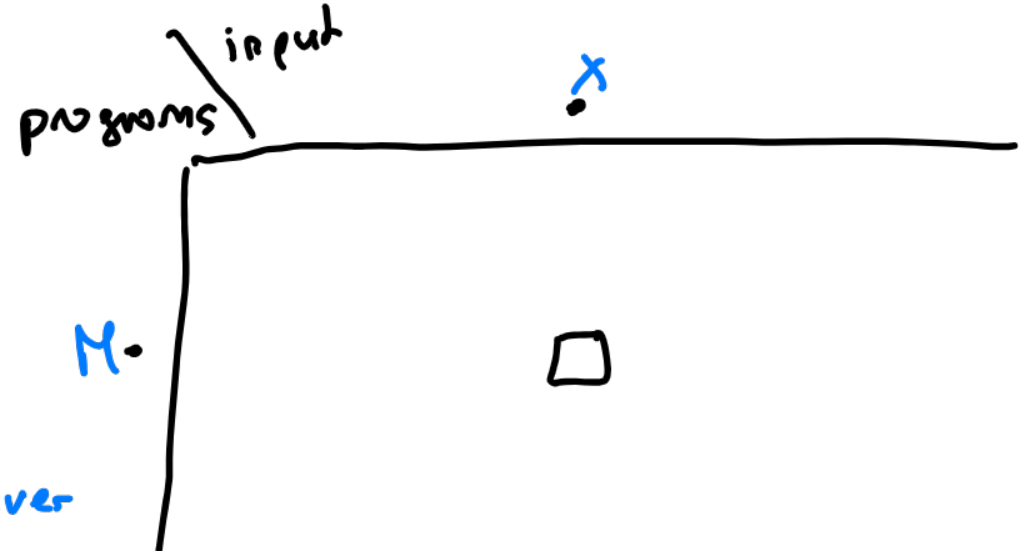
Program, written in ASCI := sequence of 0/1s

Claim: possible 0/1 sequences are countable

Proof: 0, 1, 00, 10, 01, 11,
list by length (short ones first)

Claim 2: possible input are 0/1 sequences, also countable

Create table



entry
(M, x) = 1 if
M halts on x
& 0 if it runs forever

if M not Python
or x is not
valid input for
M, write 1

Halting table of outcomes

Assume halting problem is decidable

- Algorithm Halt-decides (M, x) decides if M halts on x

$\Rightarrow$ allows us to compute any entry of this table

New program: diagonalizer

in input x

consider which input on our table x is

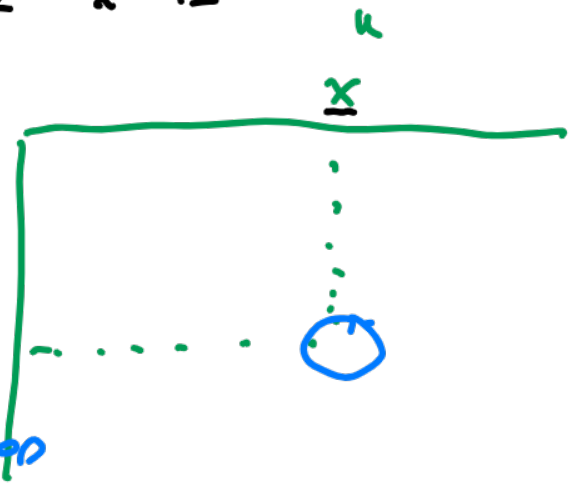
input n

consider program M in position n

Halt-decides (M, x)

if M halts on x

diagonalizer enters infinite loop



If M on x runs forever
diagonalizer terminates

Proof co-Halting not recognizable

Where is diagonalizer on the list of programs?

if not there: reason:
program in n^{th} row is not diagonalizer
as it not correct on the n^{th} input.

contradiction!

Comment: - inputs & programs are countable
- hence halt-decider cannot exist!

Busy beaver:

maximum # instructions a program can do
if in Python with at most n long program
not infinite program

alternately: max if Python n length of input + program combined

$B(n)$

Claim $B(n)$ cannot be computed.

Simple algorithm:
try all possible codes of n length
& see which runs longest

trouble: how to decide if M on x is infinite loop?

Proof: Halting $\leq$ computer $B(n)$

input to halting (M, x) — has length n

compute $B(n)$, simulate M on x for $B(n)$ steps
if it does not terminate $\Rightarrow$ infinite loop