COMS W3261: Theory of Computation

Tim Randolph

twrand.github.io/3261.html

Today:

1. What is this course about?
2. Course Structure
3. Syllabus
4. Strings, Languages, Concept recognition

---

5. DFAs (Deterministic Finite Automata)
6. Regular Languages

1. What is this course about?

Using math to answer fundamental questions about computation.

why math?
what math?

what is computation?

what questions?

Empirical Science

1. What's going on?

1. What's going on.
2. Look at stuff (do experiments)
3. Organize observations into explanations.
4. Hope observations are helpful/predictive.

Formal Science

1. What's going on?
2. Invent concepts and symbols.
3. Prove new conclusions.
4. Hope my conclusions help us.

TCS := formal science for computers

---

2. How can math teach us about computers?

Theorem. (Cantor, 1891). You can't enumerate the real numbers $\mathbb{R}$.

describe a finite rule 'program' for writing them out in some order.

$\mathbb{N} = \{1, 2, 3, \dots\}$

```
i = 1
while (true):
    print i
    i = i+1
```

$\mathbb{Z} = \{\dots -2, -1, 0, 1, 2, \dots\}$

```
i = 1
print 0
while true:
    print i, -i
    i = i+1
```

Proof. (Impossible for HK.)

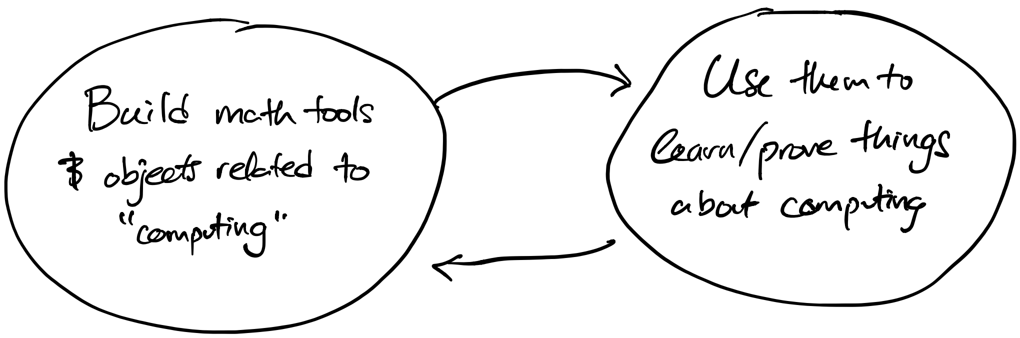
Suppose for contradiction that there *does* exist a procedure for enumerating $\mathbb{R}$. Consider the output of this procedure. For example:

$$
\begin{array}{l}
\cancel{0}.0002\cdots \\
1.\cancel{0}000\cdots \\
0.0\cancel{9}21\cdots \\
0.32\cancel{1}3\cdots \\
\vdots
\end{array}
\quad \Rightarrow \quad \underline{1.102\cdots}
$$

Create a new real by incrementing the n^{th} digit of the n^{th} number for all n.

Claim: our new real is nowhere in the original list. Contradiction. $\therefore$ $\mathbb{R}$ cannot be enumerated. $\square$

3. Course Structure



Tools:

Automata — 'concept recognizers'
- Deterministic Finite Automata (DFAs)
- Nondeterministic FA
- Pushdown Automata
- Turing Machines

Grammars — 'concept definers'
- Regular Expressions

Context-Free Grammars

Decidability — Can a computer answer this sort of YES/NO question?

Reducibility — Two tasks: task A, task B.

If you can do A, you can do B.
("Reducing" B to A.)

Complexity — If A is doable — How hard is A?

3. Syllabus.

! READ WEBSITE! !

- People. Me: Tim
  TAs: Eumin, Annie, Quintus, Elena, Brian
  OHs, dates, times,
- Links: CourseWorks, Ed, GradeScope, YouTube
- Modality: In-person, livestream (Zoom), asynchronous
- Homework: 6 problem sets.
  Assigned Monday, due Monday 11:59 PM EST.
  Can turn in: anything complete, neat, readable.
  LaTeX: Overleaf to get started.
- Exam: cumulative, virtual. Any consecutive 12h

within a certain 10-hour period. (August 10-11)

— Breakdown: 6 p-sets × 12%
+ 1 exam × 28%

100%

— Collaboration:

P-sets: open textbook, open notes, open reference websites.

Collaboration OK (esp. in office hours!)

↳ write up own works

↳ no sharing written sol'ns / notes.

---

4. 'Concepts' – Strings & Languages

Def. An alphabet := a nonempty, finite set.

Examples. $\{0, 1\}$ (Binary)

$\{a, b, c, \ldots, ñ, ø, \ldots z\}$ (Roman)

$\{0, 1, \ldots 9\}$ (Decimal)

{○, □, ★, ☺}

symbols / characters

Def. A string := finite sequence of symbols from an alphabet.

ordered set

Examples. 010, 0, 11111 (0, 1, 0)

'cat', 'dog', 'zxbeff'

'101', '199'

ε := empty string

Given strings w, x:

$|w|$:= length of string w.

w^R := reverse of w. $cat^R \to tac$

wx := concatenation 101199

Def. "Lexicographic order" := alphabetic or dictionary order.
Sort by first symbol, then next symbol, and so on; short strings first.

cat, dog, a, aa, ab, ε, pizza

$\Downarrow$

ε, a, aa, ab, cat, dog, pizza.

Sort me: $\{0, 00, 10, 111, 010\}$

Def. Language := any (possibly infinite) set of strings.

$\{0, 1, 11, 10, 111\}$

$\{0, 1\}^k \leftarrow$ pick any k symbols from $\{0, 1\}$.

$\{x \mid x \text{ contains a '0'}\}$

$\{x \mid x \text{ is the decimal representation of a prime number}\}$

$\{w \mid w \text{ is a word in the English dictionary.}\}$

Idea: Languages are like concepts.

Idea: Being able to tell if a string w is in a language L is "like" being able to recognize a concept L.

Next: Our first language-recognizing machines!