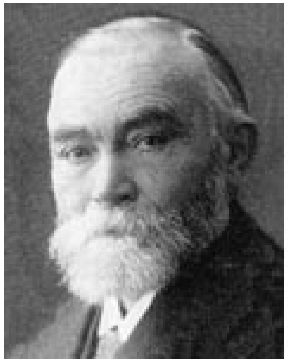
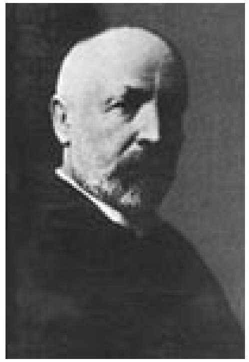
CS20a: Computation, Computers, Programs


- Instructor: Jason Hickey
  - *Email: jyh@cs.caltech.edu*
  - *Office hours: TR 10-11am*
- TAs:
  - *Nathan Gray (n8gray@cs.caltech.edu)*
  - *Brian Aydemir (emre@cs.caltech.edu)*
  - *Jason Frantz (frantz@its.caltech.edu)*
  - *TBA (tba@devnull.com)*

Administrivia

- Course texts:
  - *Kozen: Introduction to Computability*
  - *Hickey: Introduction to OCaml*
- You *must* have a CS account. Sign up at http://sysadmin.cs.caltech.edu

Resources

- All course material is posted on the course website (there are no physical handouts):
  - *http://www.cs.caltech.edu/~cs20/a*
- Mail:
  - *Send requests to course admin at cs20-admin@metaprl.org*
  - *You are encouraged to submit homework via Osaka*
    - *To submit HW1, use "cs134-submit hw1"*
    - *CS20 technical mail to jyh@cs.caltech.edu is usually ignored*
  - *General course announcements are posted to cs20-class@metaprl.org*
    - *To subscribe, send email to cs20-class-subscribe@metaprl.org*
    - *The rest of the message doesn't matter*

Prerequisites

- CS20 is typically taken 2nd year undergrad.
- CS1,2 or equivalent is required.
- Basic knowledge of algorithms and discrete data structures.
- Knowledge of some programming in a basic imperative language like C/C++, Pascal, Modula 3, Java, Oberon, Eiffel, etc.

Course work

- There are 9 homeworks, due each Monday
- One midterm and final
- Extension policy
  - *You get 8 24-hour extensions, use up to 2 on any assignment*
  - *Submit with cs20-submit, extend with cs20-extend*
  - *Extensions are only for HW/labs, not exams*

Grading policy

- Grade distribution
  - *Homework/Labs: 50%, Midterm: 20%, Final: 30%*
- Style grading
  - *Written work 10% style, labs 20% style*
- Due dates
  - *Assignments are due 11:59pm on due date*
  - *Late penalties*
    - *25% penalty up to 1 week late*
    - *50% 2 weeks late*
    - *No credit after that*
  - *There will be extra credit work*
- You can get a grading summary online

Grading policy

- Collaboration is encouraged on homeworks and labs. You may discuss the problems freely, but you must write your code and homework submissions yourself. See the Policy page on the course web site.
- Solutions are posted on the due date. Do not look at them if you have not completed the assignment.

Laboratory assignments

- Typically, each homework will include a lab component.
- For labs, we will be using the OCaml programming language
  - *You can work at home, or the machines in JRG 154*
  - *Lab submissions must be online using Osaka grader*
- HW/Lab assignments will take roughly 6 hours/week.
- OCaml
  - *If you like, you may take CS11 concurrently*

Year outline

- CS20a: computability
  - *What is computable?*
- CS20b: complexity
  - *What is effectively computable?*
- CS20c:
  - *TBA*

CS20: Fundamental concepts in CS

- The CS20 series is primarily a theory course
  - *You will learn fundamental concepts like:*
  - *Computability:*
    - *what is computable, what is hard to compute?*
  - *Computational models*
    - *Automata, Turing machines, recursive functions, lambda calculus, …*
  - *Programs*
    - *What is a program or specification?*
    - *When are two programs equal?*
    - *What is a proof?*

CS20a: Computability

- Automata theory
- Formal languages
- Program equivalence
- Basic computational models
  - *Automata*
  - *Recursive functions*
  - *Lambda calculus*
- Turing machines
  - *Recursive, recursively-enumerable sets*

CS20a: Course Outline

- Regular languages
  - *Deterministic Finite Automata (DFAs)*
  - *Nondeterministic Finite Automata (NFAs)*
- Context-free languages
  - *Pushdown automata (non/deterministic) (PDAs)*
- Recursive, R.E. languages
  - *Turing machines*

Basic set theory

- We take the naïve point of view regarding sets
  - *The meaning of the word "set" is intuitively obvious*
  - *We will deal with sets in an intuitive manner*
    - *Careless use of set theory can lead to contradiction*
    - *Russell's paradox: what is the set of all sets that do not contain themselves?*
- Notation
  - *Lowercase letters stand for elements*
  - *Uppercase letters stand for sets*
  - $a \quad A$

Set predicates, construction

- a is a member of A: $a \in A$
- set A is a subset of set B: $A \subseteq B$
- set A is a proper subset of set B: $A \subsetneq B$
- set A has elements a, b, c: $A = \{a, b, c\}$
- *Comprehension* is used to collect all the elements of a set A that have some property P

  $B = \{i \mid i \text{ is an even integer}\}$

Basic set operations

- The symbol $\vee$ means *or* (and it is inclusive)
- The symbol $\wedge$ means *and*
- $A \cup B \equiv \{x \mid x \in A \vee x \in B\}$
- $A \cap B \equiv \{x \mid x \in A \wedge x \in B\}$
- The power set: $2^A \equiv \{B \mid B \subseteq A\}$
- Difference $A - B \equiv \{x \mid x \in A \wedge x \notin B\}$

Cartesian product

- $A \times B \equiv \{(a, b) \mid a \in A \wedge b \in B\}$
- A *relation* R on a set A is a set of ordered pairs from A: $R \subseteq A \times A$
- We often write $a_1 \; R \; a_2$ for $(a_1, a_2) \in R$
- For example, consider the relation

  $< \equiv \{(0,1),(0,2),\ldots,(1,2),(1,3),\ldots\}$.

  We write $i < j$ if $(i, j) \in <$.

Orders

- An *equivalence* relation R on a set A has the following properties:
  - For each $x \in A$, $x \; R \; x$.
  - If $x \; R \; y$, then $y \; R \; x$.
  - If $x \; R \; y$ and $y \; R \; z$, then $x \; R \; z$.
- A *total order* R on a set A is a relation with the following properties:
  - If $x \neq y$, then $x \; R \; y$ or $y \; R \; x$.
  - For no $x \in A$ does $x \; R \; x$ hold.
  - If $x \; R \; y$, and $y \; R \; z$, then $x \; R \; z$.

Functions

- A function $f : A \to B$ is a relation ($f \subseteq A \times B$), and it is single-valued:

  If $(a_1, b_1) \in f$ and $(a_2, b_2) \in f$ and $a_1 = a_2$, then $b_1 = b_2$.

- An *injection*: if $f(a_1) = f(a_2)$, then $a_1 = a_2$.
- A *surjection*: for each $b \in B$, there is an a such that $f(a) = b$.
- A *bijection* is an injection and a surjection.

Examples

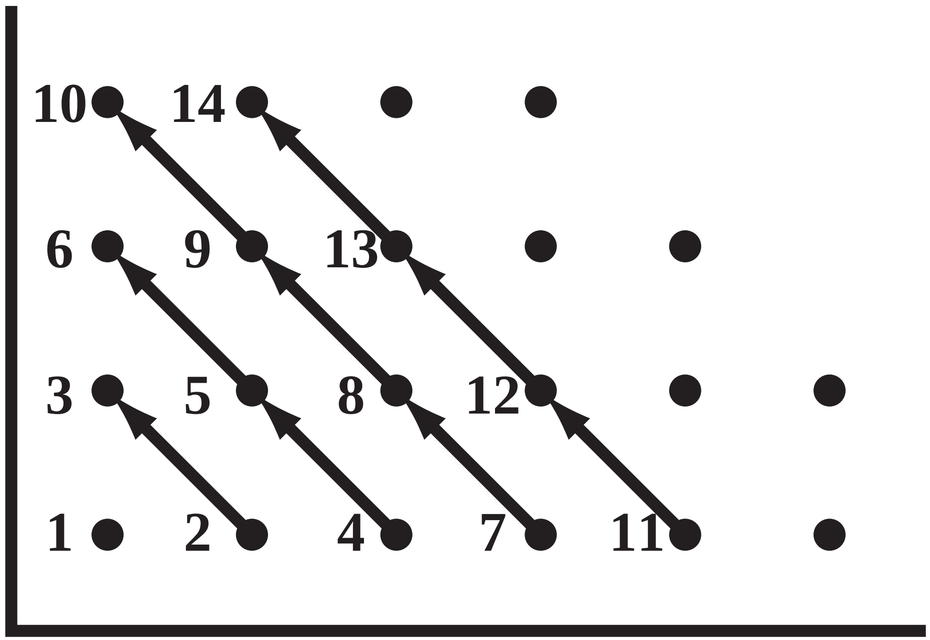
- Natural numbers: $\mathbb{N} \equiv \{0, 1, 2, 3, \ldots\}$
- Real numbers: $\mathbb{R}$
- Nonnegative reals: $\mathbb{R}^+ \equiv \{x \in \mathbb{R} \mid x \geq 0\}$
- $f(x) = x^2$ is neither an injection nor a surjection on $\mathbb{R} \to \mathbb{R}$
- $f(x) = x^2$ is an injection on $\mathbb{N} \to \mathbb{N}$
- $f(x) = x^2$ is a bijection on $\mathbb{R}^+ \to \mathbb{R}^+$

Countable sets

- A set A is *finite* if there is a bijection $A \to \{1, \ldots, n\}$ for some $n \in \mathbb{N}$.
- A set is *infinite* if it is not finite.
- A set A is *countable* if it is finite, or it is infinite and there is a bijection $f : \mathbb{N} \to A$.
- A set is *uncountable* if it is not countable.

Countable examples

Theorem The set $\mathbb{N} \times \mathbb{N}$ is countable.
Proof (by picture)



Countable examples

Real proof
Let $f(x, y) = (x + y - 1, y)$.
Let $g(x, y) = (x - 1)x/2 + y$.
Then $g \circ f$ is a bijection $\mathbb{N} \times \mathbb{N} \to \mathbb{N}$.

Corollary The set of rationals $\mathbb{Q}$ is countable.

Reals are uncountable (Cantor)

Theorem The real numbers $\mathbb{R}$ are uncountable.
Proof

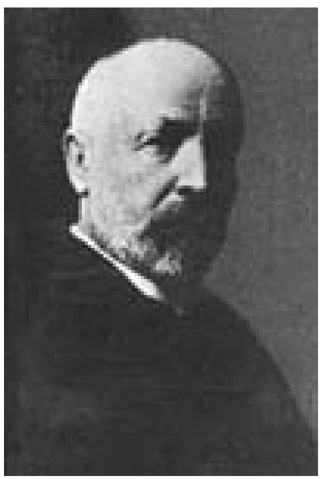
1. Assume $[0, 1)$ is countable, and enumerate the interval.

2. Construct a new number by selecting digits along the diagonal, where if the n^{th} digit of the n^{th} number is i, the n^{th} digit of the constructed number is some digit other than i and not 0 and not 9.

3. The constructed number is not in the list.

Reals are uncountable

- Cantor's diagonalization argument



0	0.	7	3	5	9	2...
1	0.	3	7	7	1	8...
2	0.	0	0	1	0	0...
3	0.	9	7	9	1	4...
4	0.	8	1	5	2	5...
5	0.	6	6	6	0	0...

0. 4 1 8 3 1...

Countability in the plane

- It is possible to draw an uncountable number of circles on the plane
  - *For each real number x, draw a circle of radius x about the origin*

What about non-intersecting figure-8s?

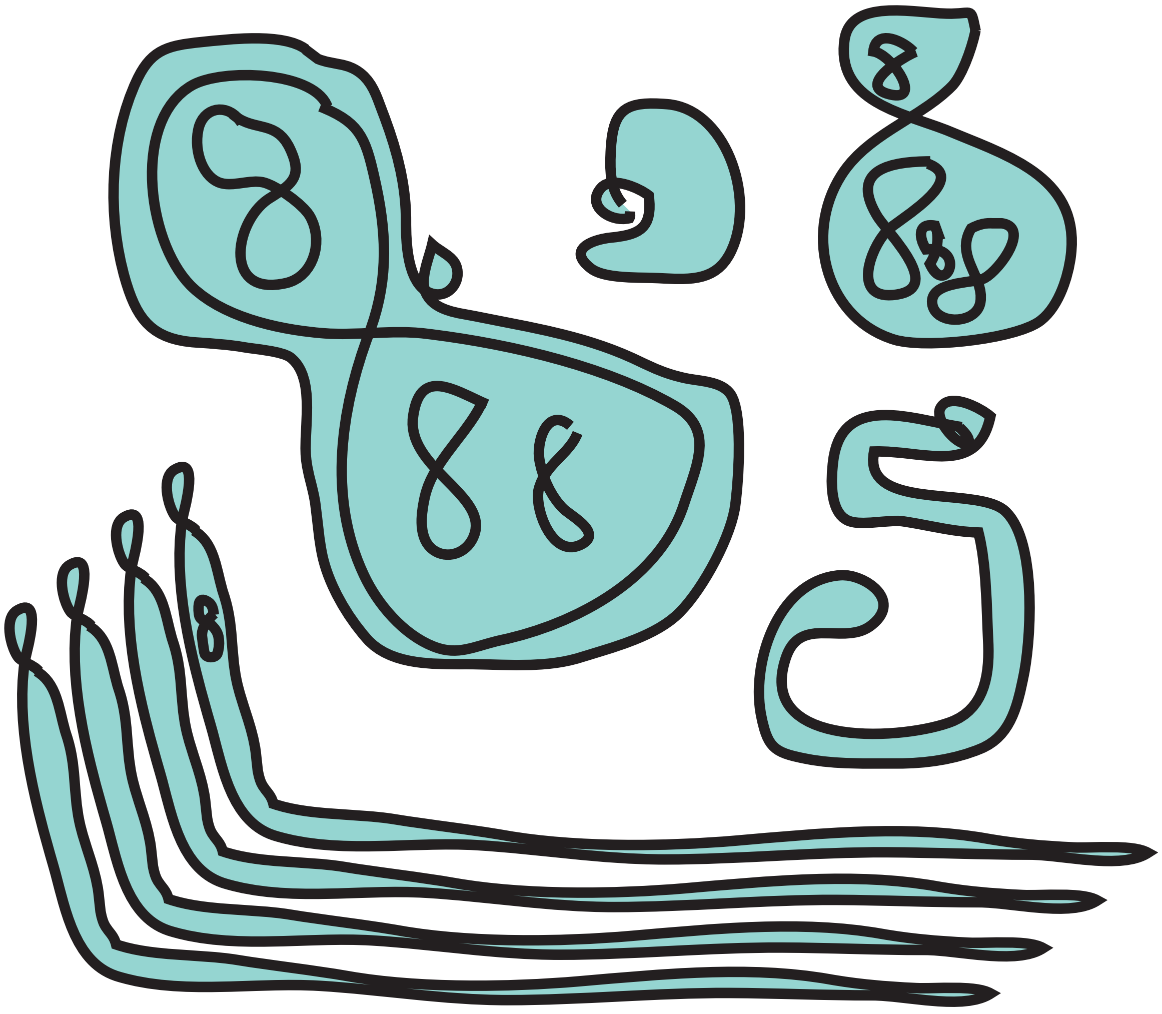


Figure-8s

Theorem Only a countable number of non-intersecting figure-8s can be drawn on the plane.

(Semi) proof

1. The real numbers $\mathbb{R}$ and the rational numbers $\mathbb{Q}$ are dense: for any two numbers $a < b$, there is a third such that $a < c < b$.

2. Each figure-8 can be uniquely identified by choosing two rational numbers, one in each loop.

3. Since $\mathbb{Q}$ is countable, so is $\mathbb{Q} \times \mathbb{Q}$, so the number of figure-8s is countable.