Lecture 1: Introduction

CSC 320: Foundations of Computer Science

Quinton Yong

quintonyong@ uvic.ca

University of Victoria

Lectures and Tutorials

- **Instructor:** Quinton Yong
  - Email: quintonyong@uvic.ca (please use your UVic email)
  - Office: ECS 621

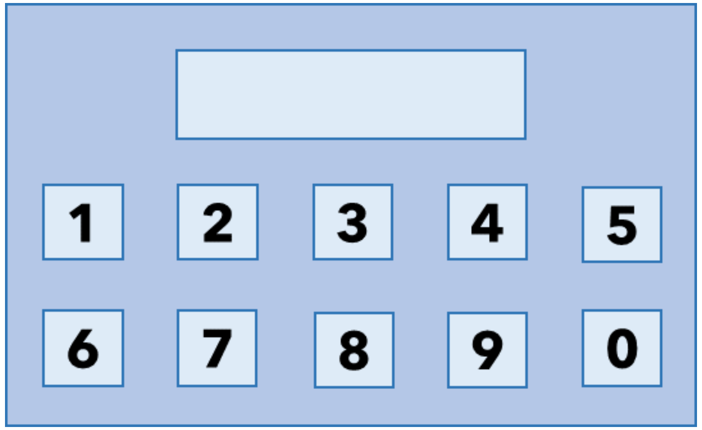
- **Lectures:**
  - Synchronous, in-person delivery
  - A01, A02: 11:30 am – 12:50 pm on M, Th (HHB 105)

- **Office Hours (ECS 621):**
  - M, Th: 1:00 pm – 2:30 pm
  - By appointment
  - Extra hours for assignments / exams

- **Course Website:**
  - Brightspace: https://bright.uvic.ca/d2l/home/336697
  - Course outline: https://heat.csc.uvic.ca/coview/course/2024011/CSC320

PowerPoint Passcode Game

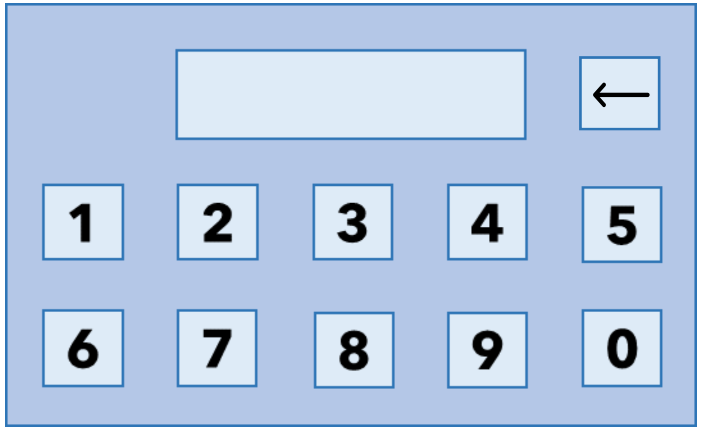
Try and guess the 3-digit passcode:



- The game is "implemented" by having **each button link to a different slide**
- How many slides are needed?

PowerPoint Passcode Game

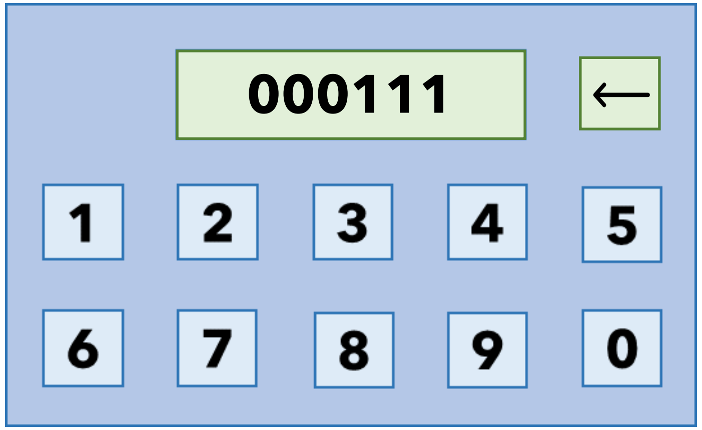
- Suppose we added an **enter button** to enter a passcode
- We want the passcode **0's followed by an equal number of 1's** (e.g 000111)



- How can we create a DFA which accepts these passcodes?

Pushdown Automata

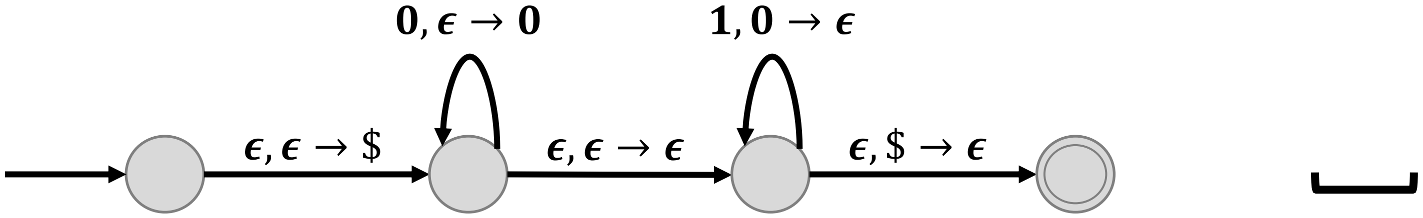
- Increase the computational power of a DFA by **adding stack memory**
- **Pushdown Automata:** Can push / pop symbols and read the top symbol



- Using a pushdown automata, we can accept passcodes of form **0's followed by an equal number of 1's**

Pushdown Automata

- Pushdown automata for **0's followed by an equal number of 1's** passcode (we will learn this notation later in the course):

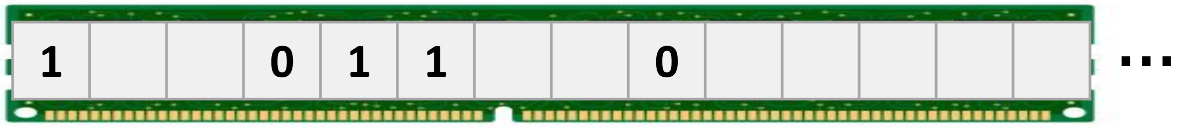


- With pushdown automata, we can determine if a text file contains **proper syntax for code** in a programming language

- There are still limits to what is computable

Turing Machines

- We give a state machine unlimited memory and unlimited read / write access
- **Turing machine:** infinite tape and can read / write symbols anywhere
- A Turing machine is an abstract **computational model** for a **classical computer**
- The computational limits of a Turing machine are the limits of classical computers



- There are problems that are **unsolvable / undecidable** on a classical computer
  - The Halting Problem
  - The Bugged Code Problem

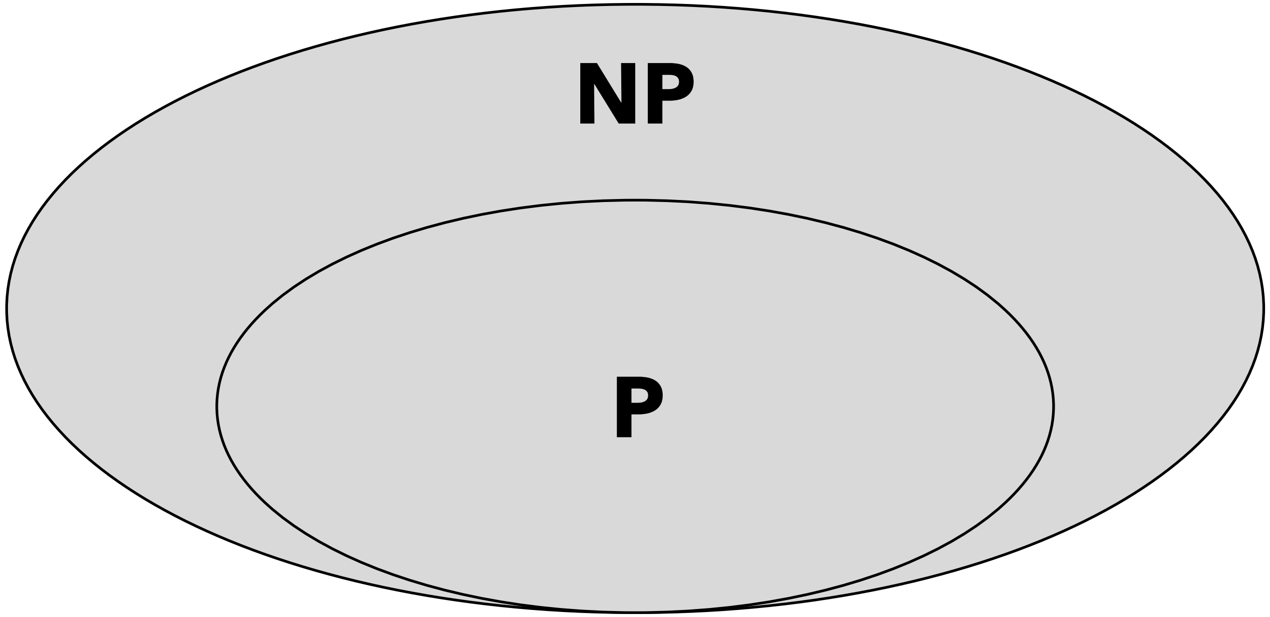
P v.s NP

For problems which are solvable on a Turing machine:

- There are problems which can be **solved efficiently**
- There are also problems which **we don't know** if there exist efficient solutions

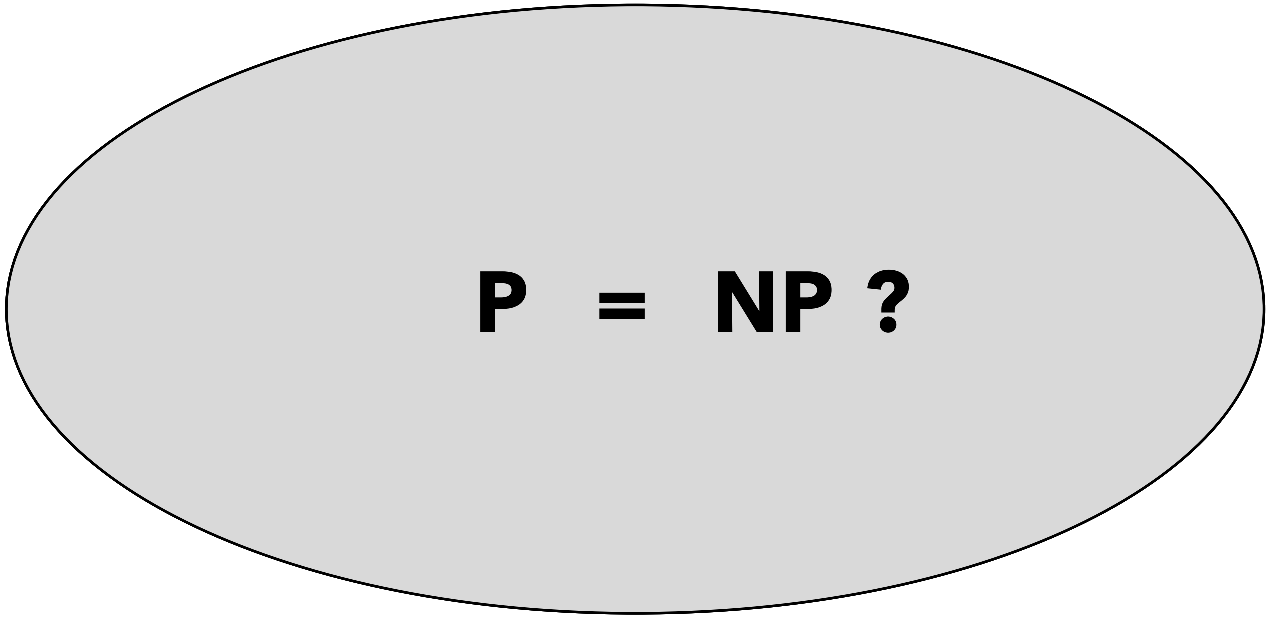
P: Problems which have polynomial time solutions (multiplication, sorting, etc.)

NP: Problems which, given a potential solution, can be verified in polynomial time

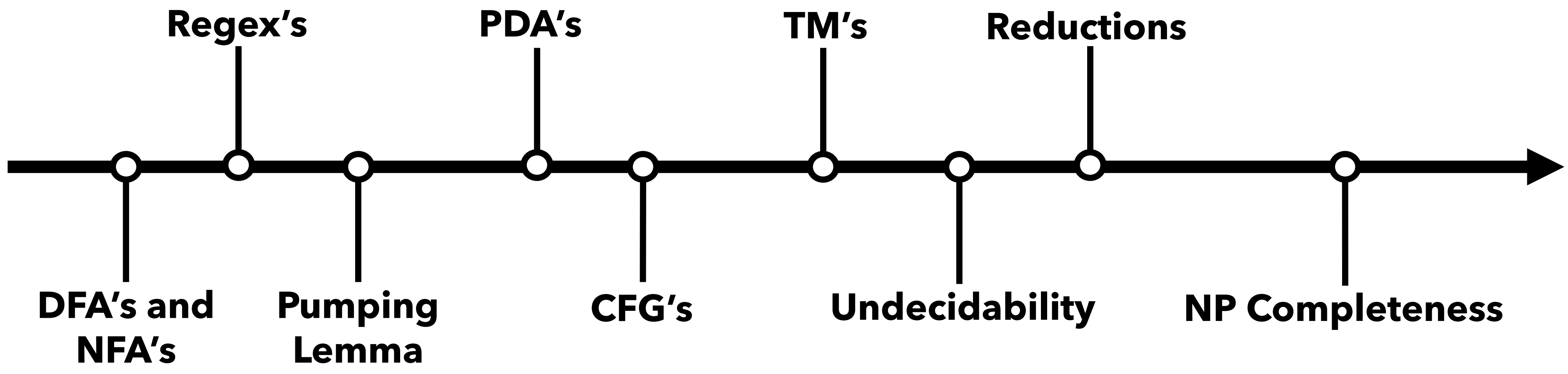


P v.s NP

- The **P v.s NP** problem: “Are the problems in P the same as the problems in NP?”
- In other words, if the solution to a problem is easy to check for correctness, must the problem be easy to solve?

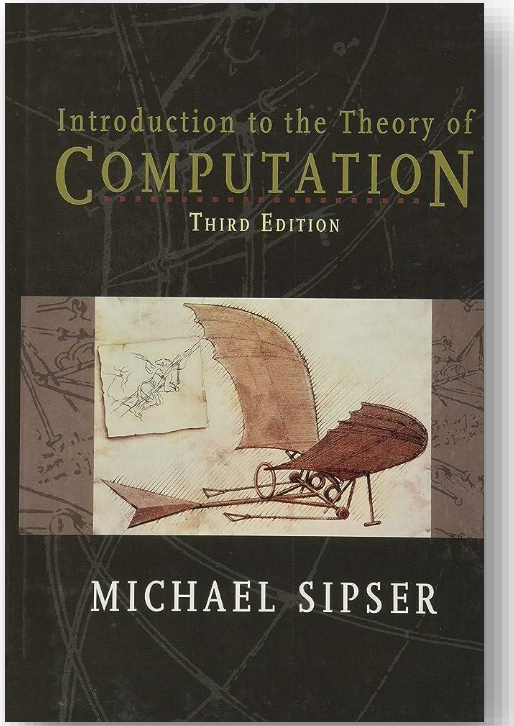


CSC 320 Timeline



Textbook (Required)

- **Introduction to the Theory of Computation, 3th Edition**
  Michael Sipster



Lectures

- **All slides presented in class will be posted**
- **Lectures not recorded live**
  - A video covering lecture content will be posted later on
  - Videos are meant for review if you miss a lecture and to supplement studying, but **not intended to replace attendance to lectures**
- **Please ask questions if you have them <u>at any point</u>**
  - If something is confusing, it is my fault for not explaining it clearly and I will gladly explain again
  - More than likely, other students have the exact same question

Tutorials

- **Weekly tutorials** going over practice questions which are similar to assignment / midterm questions

Evaluation

- **Assignments (30%)**
  - There will be **6 assignments** worth 5% each
  - You will be given around 2 weeks to complete them
  - There are 2 assignments before each midterm for practice
  - Assignments will be given and submitted through BrightSpace
- **Midterms (40%)**
  - Midterm 1 (20%): in class on **February 8th**
  - Midterm 2 (20%): in class on **March 14th**
  - You are allowed one single sided handwritten cheatsheet of A4 (8.5″ x 11″) paper
- **Final (30%)**
  - To be scheduled by the University
  - You must pass the final exam to pass the course

Policy on Collaboration / Online Resources

Assignments:

- Students are encouraged to discuss assignments together
- All solutions must be **individually written** and **no sharing of any solutions**
- (Don't look at any other student's paper)
- You may use online resources to help you on your assignments, but your submission must clearly display that you understand the solution

ChatGPT:

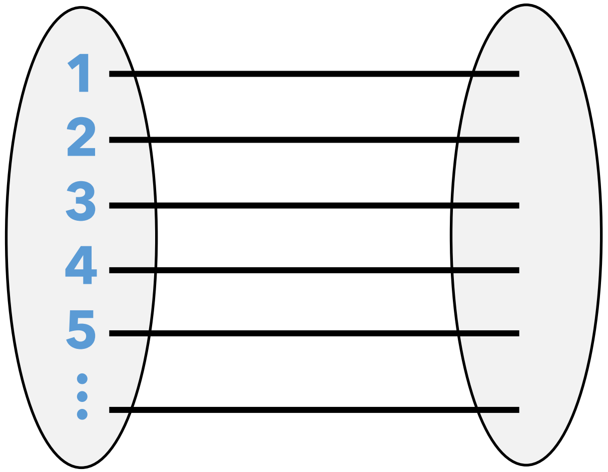
- You may use ChatGPT to help you on your assignments
- WARNING: ChatGPT is pretty bad at CSC320…

Countable and Uncountable

A set is **countable** if it is **finite** or **countably infinite**

- elements of a countable set can be counted one at a time without missing any
- every element is associated with a unique natural number

There exists a **bijection** between any **countably infinite set** and the **set of natural numbers** $\mathbb{N}$



A set that is neither finite nor countably infinite is **uncountable**

Countable and Uncountable

Is the set of **integers** $\mathbb{Z}$ countable?

$$\ldots\ -5\ \ -4\ \ -3\ \ -2\ \ -1\ \ 0\ \ 1\ \ 2\ \ 3\ \ 4\ \ 5\ \ \ldots$$

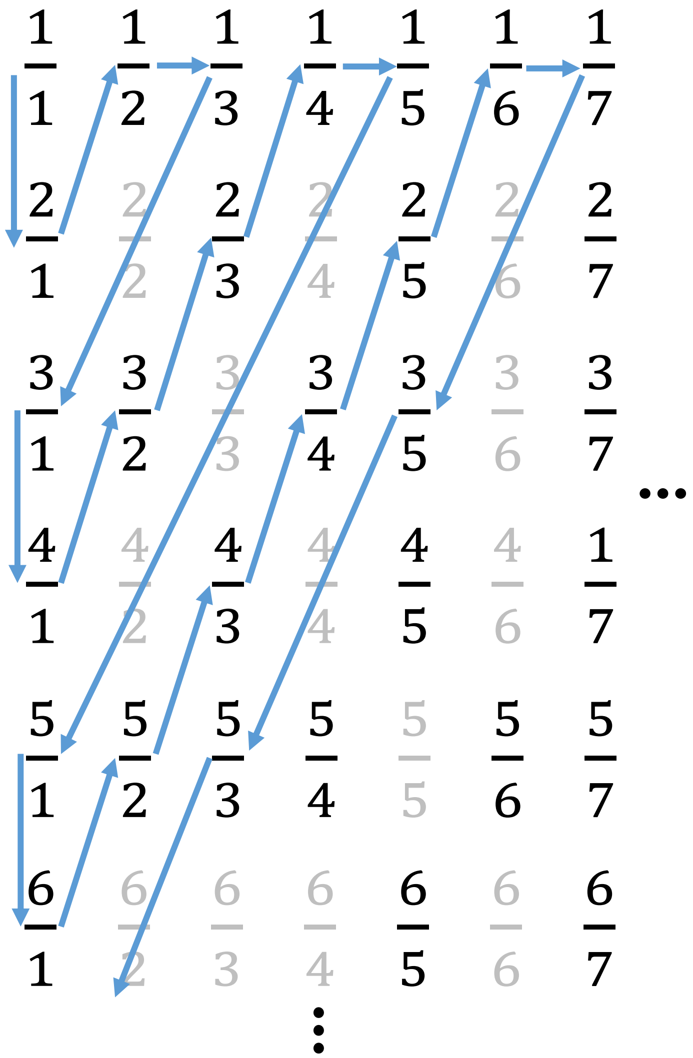
We can enumerate the elements of $\mathbb{Z}$ as follows:

$$\begin{array}{ccccccccccccc} 0 & 1 & -1 & 2 & -2 & 3 & -3 & 4 & -4 & 5 & -5 & \ldots \\ 1 & 2 & 3 & 4 & 5 & 6 & \ldots & & & & & \end{array}$$

The set of integers $\mathbb{Z}$ is **countably infinite**

Countable and Uncountable

Is the set of **positive nonzero rational numbers** $\mathbb{Q}^{+}\backslash\{0\}$ countable?



- This method of counting lets us enumerate all rational numbers
- No missing numbers or getting stuck in infinity

- **Note:** Counting row by row or column by column would never reach all the numbers

$\mathbb{R}$ is uncountable (Cantor's Diagonalization)

Proof by contradiction: Assume that the real numbers $\mathbb{R}$ is countable.

- If $\mathbb{R}$ is countable, then we should be able to enumerate the real numbers **just between 0 and 1**.
- Let the enumeration $(x_1, x_2, x_3, \ldots)$ be written as follows:

$$
\begin{aligned}
x_1 &= 0.d_{11}d_{12}d_{13}d_{14}\ldots \\
x_2 &= 0.d_{21}d_{22}d_{23}d_{24}\ldots \\
x_3 &= 0.d_{31}d_{32}d_{33}d_{34}\ldots \\
x_4 &= 0.d_{41}d_{42}d_{43}d_{44}\ldots \\
&\vdots
\end{aligned}
$$

- $x_n = 0.d_{n1}d_{n2}d_{n3}d_{n4}\ldots$ is the n^{th} real number in the enumeration
- x_n has decimal digits $0.d_{n1}d_{n2}d_{n3}d_{n4}$ (since we are enumerating real numbers between 0 and 1)

$\mathbb{R}$ is uncountable (Cantor's Diagonalization)

- Consider the number $\boldsymbol{c} = 0.c_1c_2c_3c_4 \ldots$ where $c_i \neq d_{ii}$ for each i

$\boldsymbol{x_1} = 0.d_{11}d_{12}d_{13}d_{14} \ldots$ $\quad$ $\boldsymbol{c} \neq \boldsymbol{x_1}$ since the 1^{st} decimal digit is different ($c_1 \neq d_{11}$)

$\boldsymbol{x_2} = 0.d_{21}d_{22}d_{23}d_{24} \ldots$ $\quad$ $\boldsymbol{c} \neq \boldsymbol{x_2}$ since the 2^{nd} decimal digit is different ($c_2 \neq d_{22}$)

$\boldsymbol{x_3} = 0.d_{31}d_{32}d_{33}d_{34} \ldots$ $\quad$ $\boldsymbol{c} \neq \boldsymbol{x_3}$ since the 3^{rd} decimal digit is different ($c_3 \neq d_{33}$)

$\boldsymbol{x_4} = 0.d_{41}d_{42}d_{43}d_{44} \ldots$ $\quad$ $\boldsymbol{c} \neq \boldsymbol{x_4}$ since the 4^{th} decimal digit is different ($c_4 \neq d_{44}$)

$\vdots$

$\boldsymbol{x_n} = 0.d_{n1}d_{n2}d_{n3}d_{n4} \ldots$ $\quad$ $\boldsymbol{c} \neq \boldsymbol{x_n}$ since the n^{th} decimal digit is different ($c_n \neq d_{nn}$)

$\vdots$

- Since $\boldsymbol{c}$ is a number between 0 and 1, it **should be enumerated** in this list
- However, since it **differs from every element**, it cannot be in this list

Clarification on c

- Consider the number $c = 0.c_1c_2c_3c_4 \ldots$ where $c_i \neq d_{ii}$ for each i
- For example, suppose the numbers $(x_1, x_2, x_3, \ldots)$ are as follows

$$x_1 = 0.\mathbf{4}031\ldots$$
$$x_2 = 0.1\mathbf{8}93\ldots$$
$$x_3 = 0.53\mathbf{6}7\ldots$$
$$\vdots$$

- We define $c = 0.c_1c_2c_3c_4$ such that the digit c_i is something different than the i^{th} digit of x_i

- In the example enumeration above:
  - c_1 can be any number other than 4
  - c_2 can be any number other than 8
  - c_3 can be any number other than 6
  - So, c could be something like 0.597…

Clarification on c

- You may be wondering, if we enumerate the real numbers between 0 and 1 like

$$
\begin{gathered}
x_1 = 0.000\ldots00 \\
x_2 = 0.000\ldots01 \\
x_3 = 0.000\ldots02 \\
\vdots
\end{gathered}
$$

then c must be in the list somewhere.

- Consider if c appears in the list at position k, that is $x_k = c$
- However, c is defined such that digit c_k is different than the k^{th} decimal digit of x_k
- Thus, c can't possibly be in the list anywhere

$\mathbb{R}$ is uncountable (Cantor's Diagonalization)

- The enumerated list $x_1, x_2, x_3, \ldots$ **does not** contain all real numbers between 0 and 1 since it cannot contain c
- So, **we cannot enumerate** all the elements in this subset of $\mathbb{R}$ (real numbers between 0 and 1)
- This is a **contradiction** since we assumed that $\mathbb{R}$ is countable
- Therefore, $\mathbb{R}$ **is uncountable**