# CARDINALITY

DISCRETE MATHEMATICS AND THEORY 2
MARK FLORYAN

# GOALS!

1. Quick review of functions!

2. How do we use functions to compare the sizes of sets? Why might this be useful as we move forward talking about computation?

3. Do all infinite sets have the same size? What can this tell us (already) about the theory of computation?

# PART 1: QUICK REVIEW OF FUNCTIONS

# DEFINING FUNCTIONS

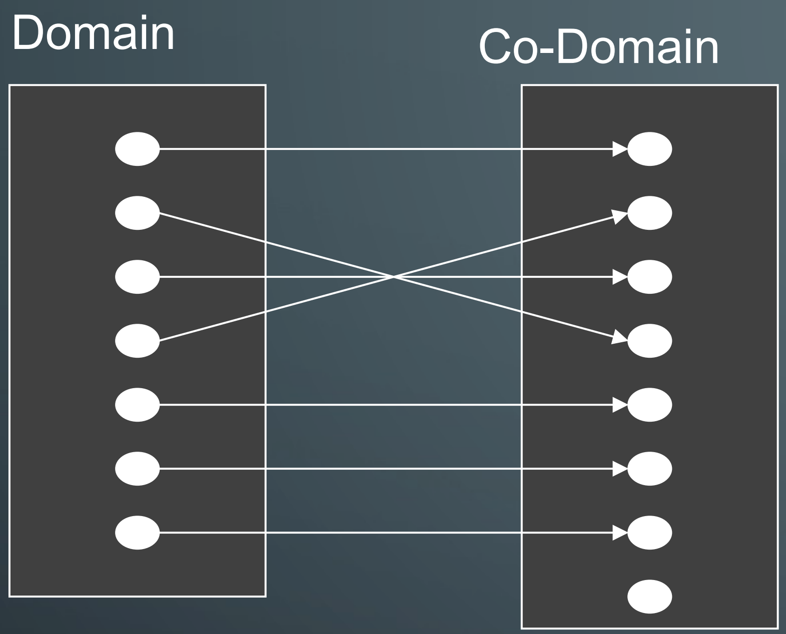
***Function***: a "mapping" from input to output

- Function  maps elements from the set  to an element from the set
- : the domain of
- : the co-domain of
- Range/image of :
  - The elements of  that are "mapped to" by something

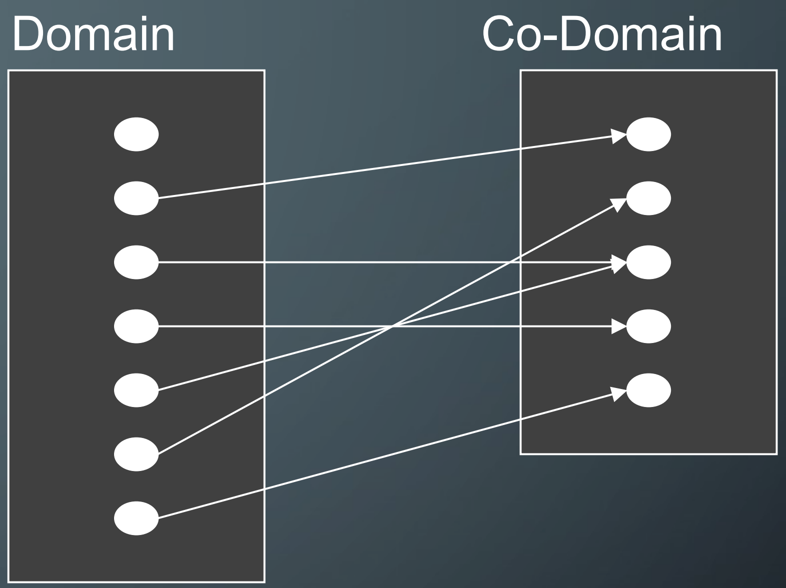
***Finite function***: a function with a finite domain

 is a finite function if  is finite. Otherwise it's an infinite function

# INJECTIVE FUNCTIONS




INJECTIVE FUNCTION




NON-INJECTIVE FUNCTION

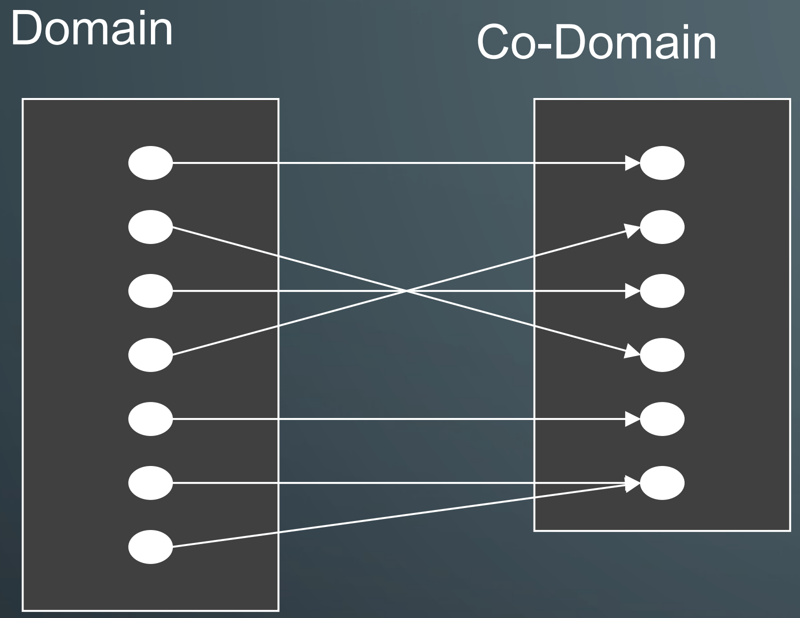
One-to-one (injective)

- Different inputs yield different outputs
- No two inputs share an output

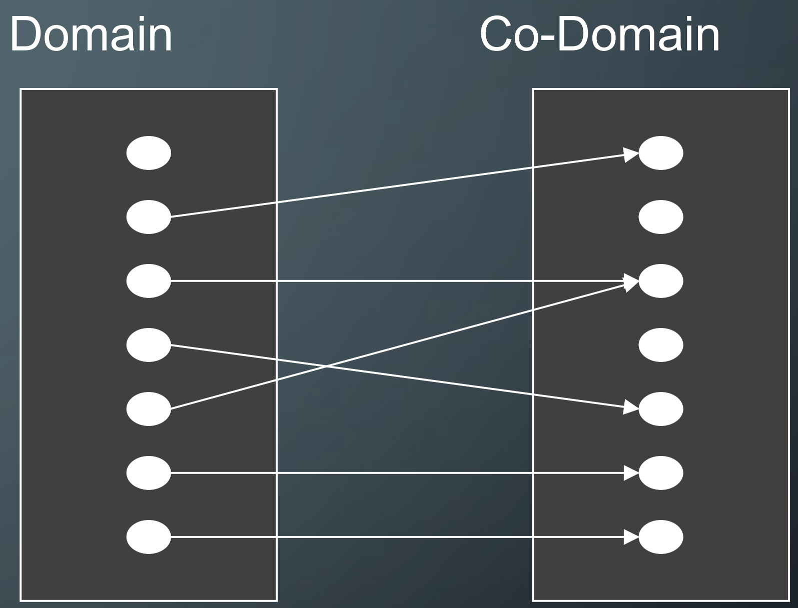
# PROPERTIES OF FUNCTIONS

- One-to-one (injective)
- Onto (surjective)
  - Everything in  is the output of something in

# ONTO, SURJECTIVE FUNCTIONS




SURJECTIVE FUNCTION




NON-SURJECTIVE FUNCTION

Everything in Co-Domain “receives” something

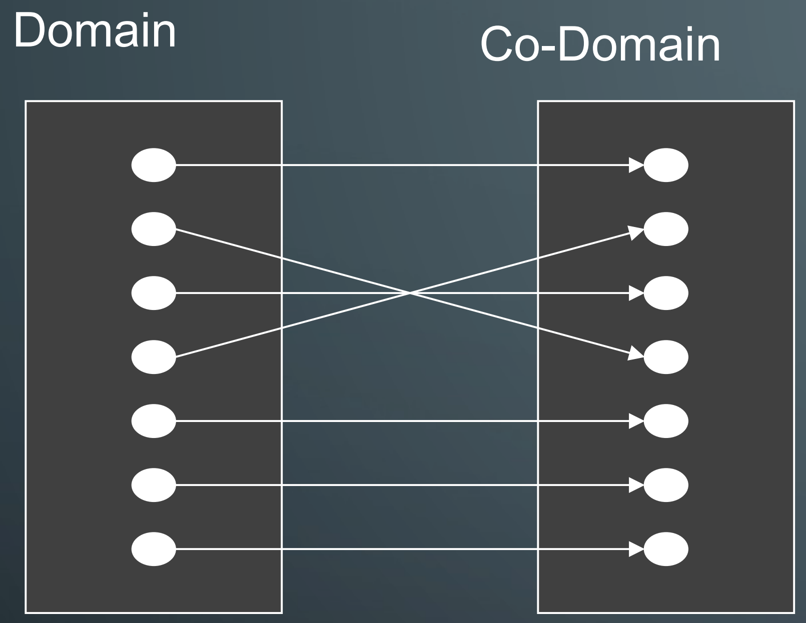
# PROPERTIES OF FUNCTIONS

- One-to-one (injective)

- Onto (surjective)

- One-to-one Correspondence (bijective)
  - Both one-to-one and surjective
  - Everything in  is mapped to by a unique element in
  - All elements from domain and co-domain are perfectly “partnered”

# BIJECTIVE FUNCTIONS




BIJECTIVE FUNCTION

Because Onto:
Everything in Co-Domain “receives” something

Because 1-1:
Nothing in Co-Domain “receives” two things

Conclusion:
Things in the Domain exactly “partner” with things in Co-Domain

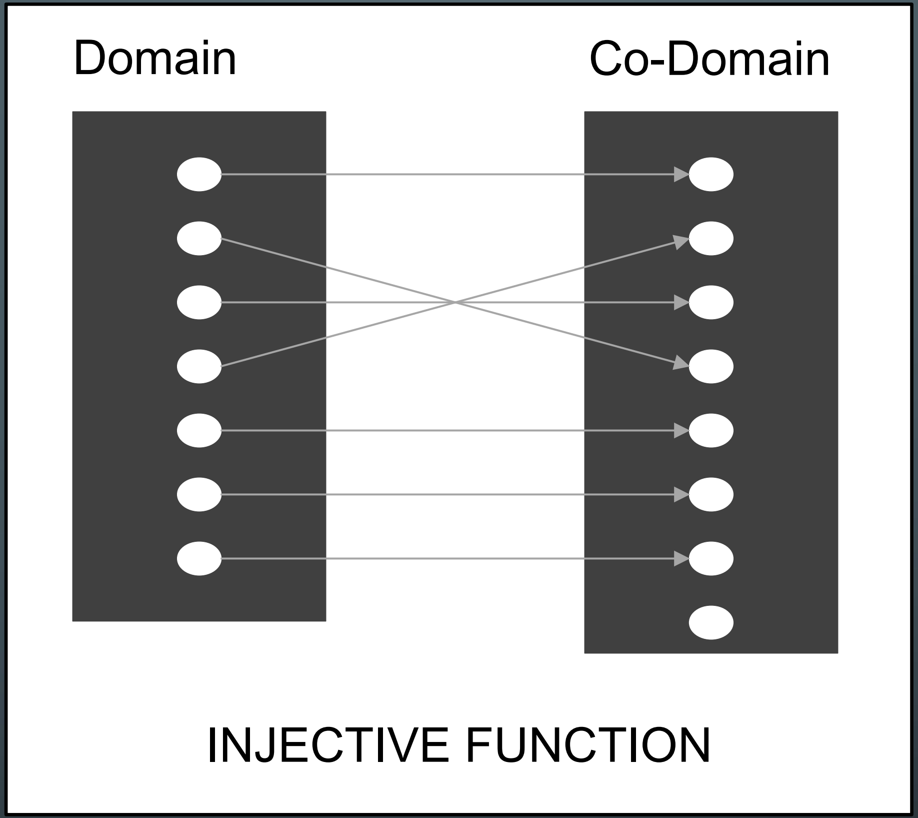
# PART 2: USING FUNCTIONS TO COMPARE SIZES OF SETS

# COMPARING CARDINALITIES WITH FUNCTIONS

- Let  be a finite function
  - 
- Consider the following possible characteristics of f
  - Injective
  - Surjective
  - Bijective

*Each of these will tell us something about the relative sizes of D and C*

# 1-1, INJECTIVE FUNCTIONS



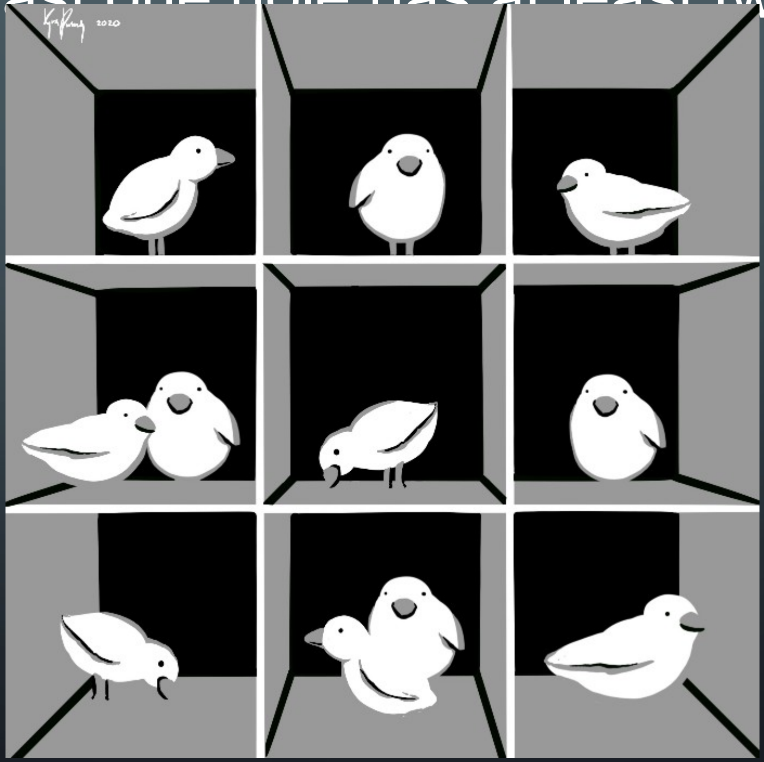

*Thus, showing there exists an injective function from D to C is one way to show that*

Nothing in Co-Domain “receives” two things

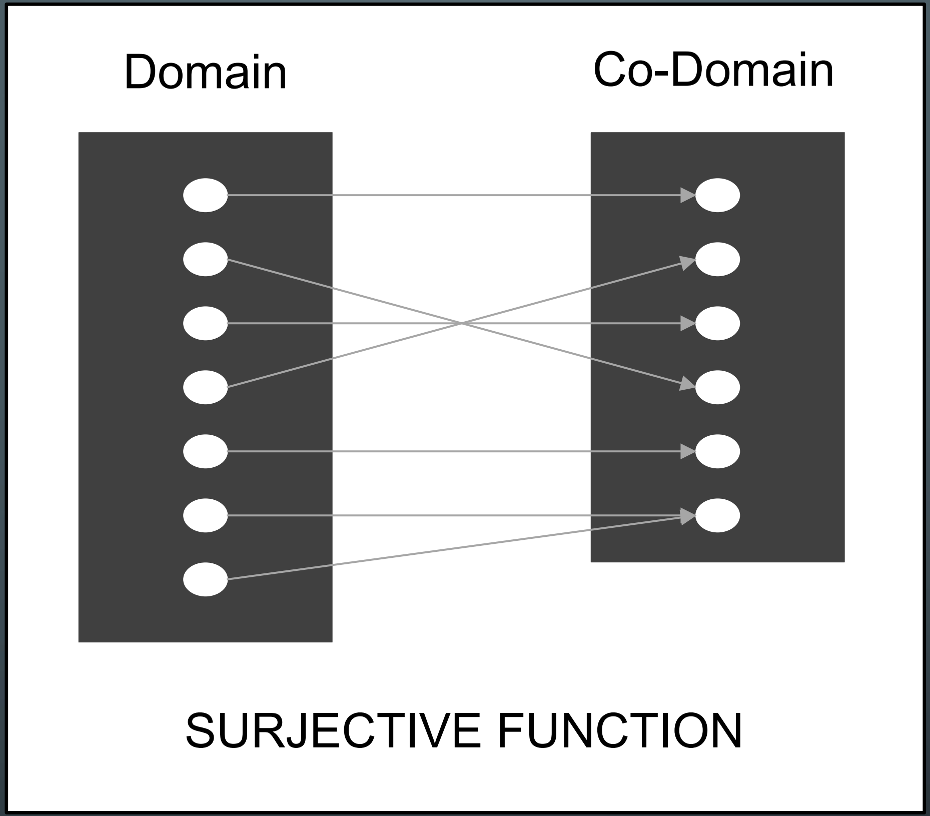
****Only possible if**

# PIGEONHOLE PRINCIPLE

- Every pigeon is sitting in a hole
- There are more pigeons than there are holes
- At least one hole has at least two pigeons



# ONTO, SURJECTIVE FUNCTIONS



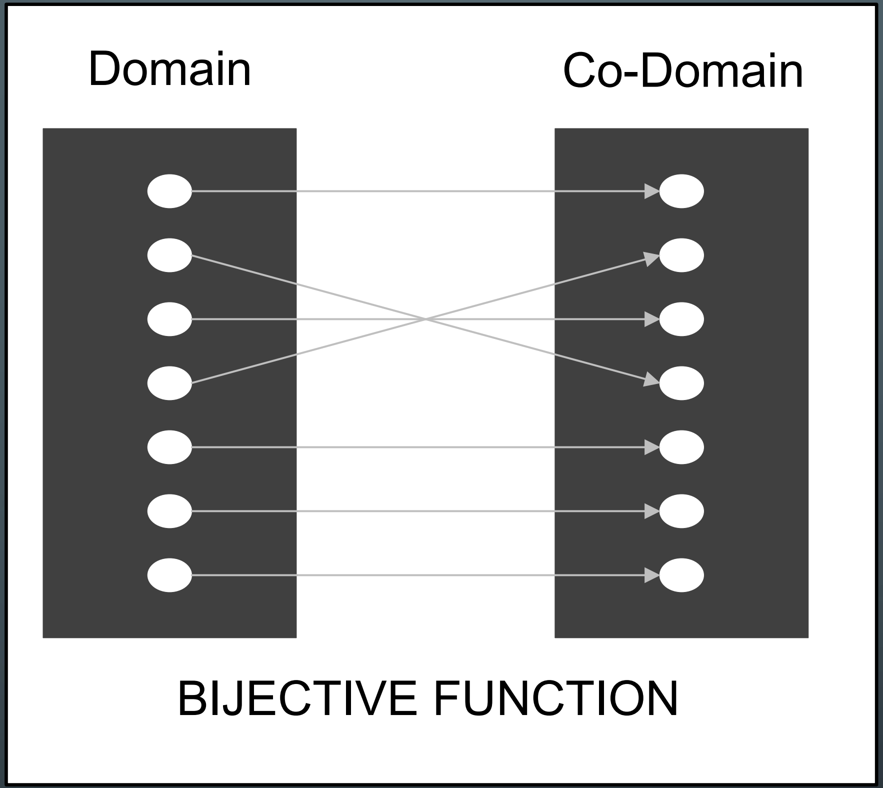

*Thus, showing there exists a surjective function from D to C is one way to show that*

Everything in Co-Domain “receives” something

****Only possible if**

# BIJECTIVE FUNCTIONS

*Because 1-1: Nothing in Co-Domain "receives" two things*




*Because Onto: Everything in Co-Domain "receives" something*

Conclusion:
Things in the Domain exactly "partner" with things in Co-Domain
***Note: This means that**

# COMPARING CARDINALITIES WITH FUNCTIONS

- To show
  - Find a surjective function
  - Find an injective function
- To show
  - Find a bijective function
  - Find both a surjective function  and an injective function

# PRACTICE:  VIA BIJECTION

Theorem:

How do we
show this?
Any ideas?

# VIA BIJECTION

- Proof idea:
  - Find a bijection

- Given , what is ?

  - E.g.
  - In other words, let each item b map to the natural number corresponding to the binary representation!!

# CALCULATING BINARY OF 13

- is odd, so last bit is
- 6 is even, so next bit is
- 3 is odd, so next bit is 1
- 1 is odd, so next bit is 1

$b=$ ¿

| | | | |
|---|---|---|---|
| 1 | 1 | 0 | 1 |

*…and fill with the last n-4 zeros to ensure there are n digits*

# PRACTICE:  VIA BIJECTION

Theorem:

Is the mapping we provided injective (Every input has unique output)? Why?

Is the mapping we provided surjective (Every value less than is covered)? Why?

# PRACTICE 2

Theorem: For a finite set ,

How do we
show this?
Any ideas?

# FOR A FINITE SET ,

- Find a function
- Example: let

- Bijection: give each value of  an index, for a particular subset of S, make the bit at that index 0 if it is absent, otherwise make it 1.

# WHY IS THIS A BIJECTION?

- Show that it's injective
  - Different subsets of  result in different strings
  - This holds because for two subsets of , call them  and , if  there must be some value  such that . This means that  is different from  at the bit associated with element .
- Show that it's surjective
  - Every string is mapped to by some subset of
  - Consider that we have some string . We can find the subset of  called such that by including the value associated with bit  in  provided that bit is 1

# PART 3: COMPARING SIZES OF INFINITE SETS

# INFINITE CARDINALITY

How do we compare the sizes of two infinite sets? Wait…do they not automatically have the same size?

# INFINITE CARDINALITY

We say that for (infinite) sets  and , that  if there is a bijection

# COUNTABILITY AND UNCOUNTABILITY

A set  is countable if

If , then  is “countably infinite”

A set  is countable if there is an onto (surjective) function from  to

Otherwise a set is uncountable.

# PRACTICE: SHOW THAT

# IS COUNTABLE

- Need to “represent” strings with naturals
- Idea: build a “list” of all strings, represent each string by its index in that list

# LISTING ALL STRINGS (BAD WAY)

can be defined as follows:

the number that  represents

Why is this function not a bijection?

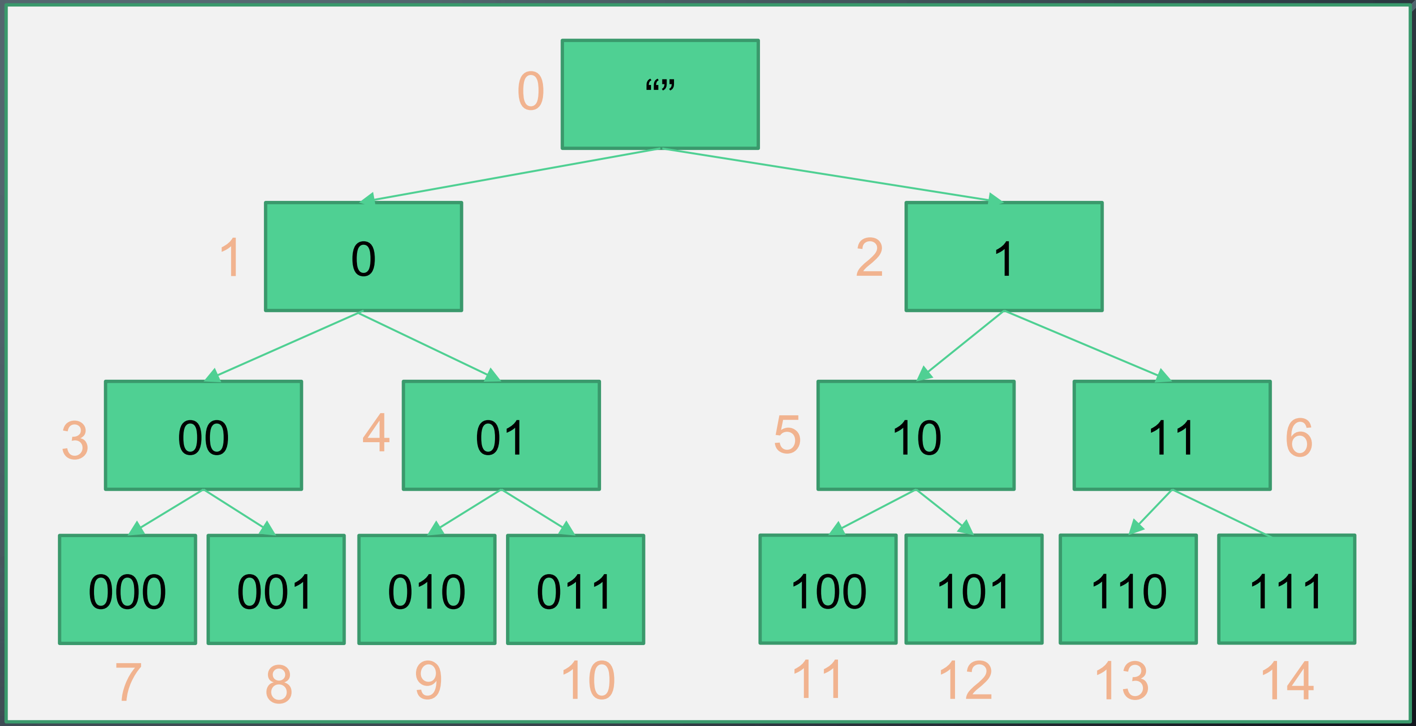
# LISTING ALL STRINGS

0

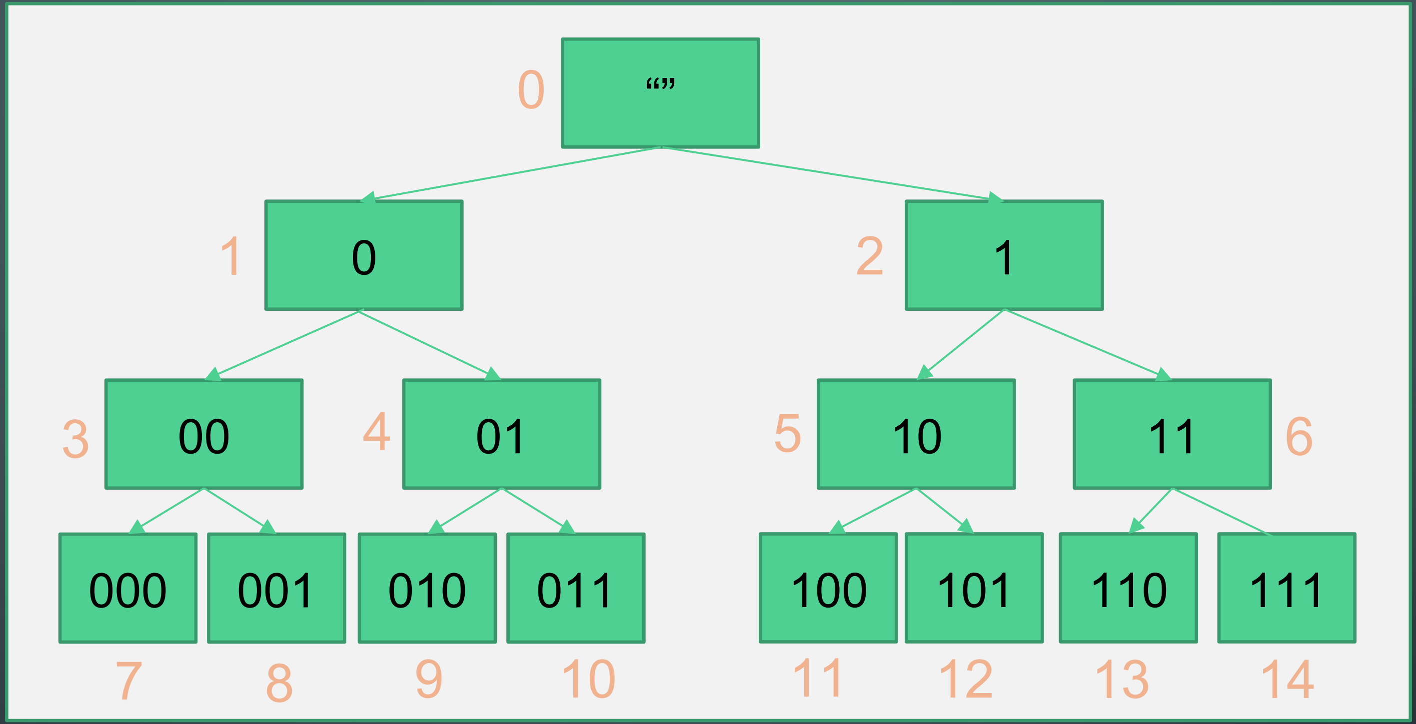
12

3 4 5 6

7 8 9 10 11 12 13 14



# LISTING ALL STRINGS

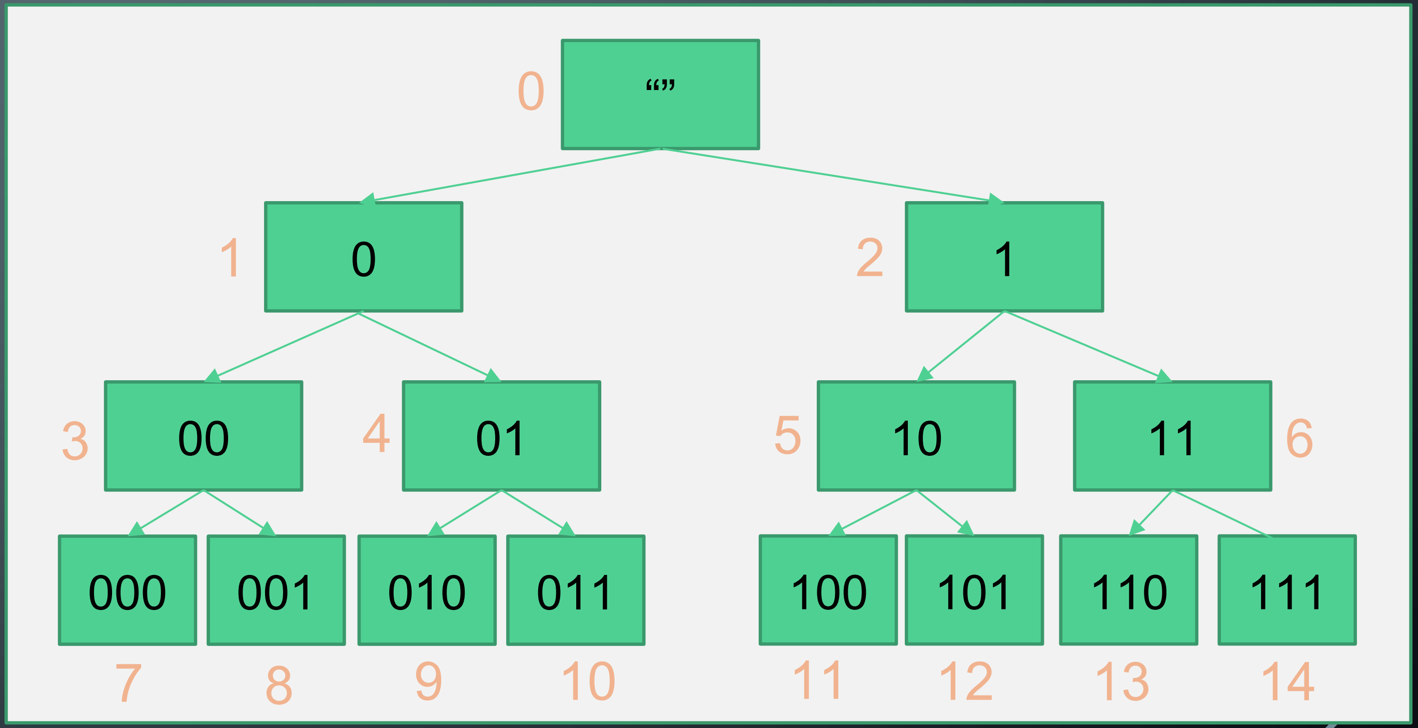



Formulaic version:

**Where b(w) is the integer value of the binary bitstring w

# WHY IS THIS A BIJECTION?

- ***Injective***: different strings map to different numbers:
  - Different strings map to different nodes in the tree
  - No two nodes in the tree have the same index
- ***Surjective***: every number appears
  - We listed them one by one and there are an infinite number of nodes.



# DEMONSTRATE THAT EACH OF THE FOLLOWING IS COUNTABLE

# PROOF:  IS COUNTABLE

$$f_{+\text{¿}:\mathbb{Z}^{+\text{¿}\leftrightarrow\mathbb{N}\text{¿}}\text{¿}}$$

# PROOF:  IS COUNTABLE

# PROOF:  IS COUNTABLE

# IS COUNTABLE

- To build
  - Idea: map natural numbers to evens, map negative numbers to odds
  - If
  - if
- Note that this means that if  and  are both countable then  is also countable!

# IS COUNTABLE

Thoughts on how to prove it?

# IS COUNTABLE

| $\mathbb{N} \downarrow$ \ $\mathbb{N} \rightarrow$ | 0 | 1 | 2 | 3 | 4 | 5 | |
|---|---|---|---|---|---|---|---|
| 0 | 0,0 | 0,1 | 0,2 | 0,3 | 0,4 | 0,5 | . . . |
| 1 | 1,0 | 1,1 | 1,2 | 1,3 | 1,4 | 1,5 | . . . |
| 2 | 2,0 | 2,1 | 2,2 | 2,3 | 2,4 | 2,5 | . . . |
| 3 | 3,0 | 3,1 | 3,2 | 3,3 | 3,4 | 3,5 | . . . |
| 4 | 4,0 | 4,1 | 4,2 | 4,3 | 4,4 | 4,5 | . . . |
| | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | |

# IS COUNTABLE

- Idea: there is a surjective mapping from  to
- This one is left as an exercise (could be on homework or quiz)

# NUMBER OF PROGRAMS AS NUMBER OF FUNCTIONS

# HOW MANY PYTHON/JAVA PROGRAMS?

- How do we represent Java/Python programs?
- How many things can we represent using that method?

# HOW MANY FUNCTIONS ?

- Short answer: Too many!
  - Uncountable

- Conclusion: Some functions cannot be computed by any java/python program
- How to prove this?

# HOW TO SHOW SOMETHING IS UNCOUNTABLE?

# UNCOUNTABLY MANY FUNCTIONS

- If we show a subset of is uncountable, then  is uncountable too
- Consider just the "yes/no" functions (decision problems):

| | |
|---|---|
| “” | 1 |
| 0 | 0 |
| 1 | 0 |
| 00 | 1 |
| 01 | 1 |
| 10 | 1 |
| 11 | 1 |
| 000 | 0 |
| 001 | 0 |

# GOAL:  IS UNCOUNTABLE

- Each function can be represented by a single infinite bitstring :  is a simpler representation of f

- Show there is no onto mapping from  to

| | |
|---|---|
| “” | 1 |
| 0 | 0 |
| 1 | 0 |
| 00 | 1 |
| 01 | 1 |
| 10 | 1 |
| 11 | 1 |
| 000 | 0 |
| 001 | 0 |

For example, this function can be fully described by the outputs only (the order of the inputs is fixed). So the right column (100111100…) fully describes this unique function

# $|\{0,1\}^{\infty}| > ¿\mathbb{N}|$

- Idea:
  - show there is no way to “list” all infinite length binary strings
  - Any list of binary strings we could ever try will be leaving out elements of

# $|\{0,1\}^{\infty}| > ¿\mathbb{N}|$

Attempt at mapping to

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| 3 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
| 4 | 1 | 0 | 1 | 1 | 0 | 1 | 0 |
| 5 | 1 | 0 | 0 | 1 | 1 | 1 | 0 |
| 6 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| … | | | | | | | |
| | 0 | 1 | 0 | 0 | 1 | 0 | 0 |

A string that our attempt missed

Derive by selecting each as the opposite of the from row

# $|\{0,1\}^\infty| > ¿\mathbb{N}|$

Attempt at mapping  to

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0 | **1** | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 0 | **0** | 0 | 0 | 0 | 0 | 0 |
| 2 | 1 | 0 | **1** | 0 | 1 | 0 | 1 |
| 3 | 1 | 1 | 0 | **1** | 1 | 0 | 1 |
| 4 | 1 | 0 | 1 | 1 | **0** | 1 | 0 |
| 5 | 1 | 0 | 0 | 1 | 1 | **1** | 0 |
| 6 | 0 | 0 | 0 | 1 | 1 | 1 | **1** |
| … | | | | | | | |
| | 0 | 1 | 0 | 0 | 1 | 0 | 0 |

Take the bolded bits across the diagonal. Select a bitstring where each of these bits is flipped. In this example: **0100100...**

# OTHER COUNTABLE/UNCOUNTABLE SETS

- Countable sets:
  - Integers
  - Rational numbers
  - Any finite set
- Uncountable Sets:
  - Real numbers
  - The power set of any infinite set

# CANTOR'S THEOREM

- For any set ,
- Even if  is infinite!
- Idea:
  - (why?)
  - There cannot be a bijection between  and
  - Not going to prove

# CONCLUSION

- There are countably many strings
  - And therefore binary strings, programs, etc.
- There are uncountably many functions
- ***<u>Some functions can’t be implemented</u>***