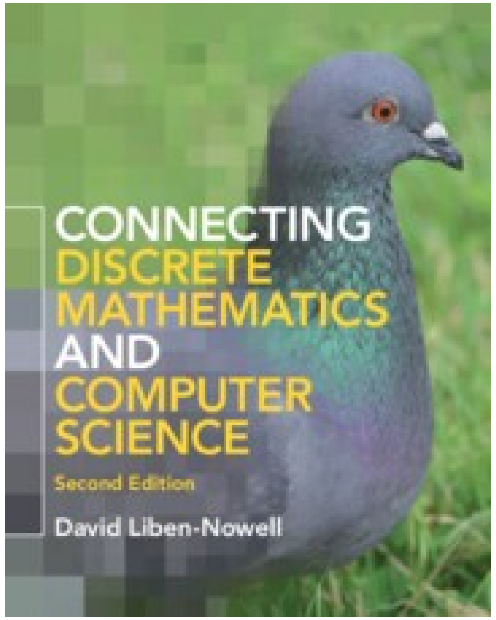
csci54 – discrete math & functional programming
basic data types: sets, function, relations



"Connecting Discrete Mathematics and Computer Science"
by David Liben-Nowell

https://cs.carleton.edu/faculty/dln/book/

- Python has 4 built-in data types for storing collections of data.
- What are they? How are they different? What is each one good for?

- <u>lists</u>: ordered, indexed, mutable, allows duplicate values
- <u>tuples</u>: ordered, indexed, immutable, allows duplicate values
- <u>dictionaries</u>: unordered, mutable, cannot have duplicate keys
- <u>sets</u>: unordered, unindexed, cannot have duplicate keys, can add/remove elements but can't change existing elements

Mathematical sets

- A set is an unordered collection of objects
- Given a set S and an object o, either $o \in S$ or $o \notin S$
- The cardinality of a set is written $|S|$ and is the number of elements in the set

- Examples of sets we've seen:
  - `Int`
  - `Integer`
  - `Char`
  - `Bool`

Z : set of integers
Z^{+}: set of positive integers
N: set of non-negative integers
Q : set of rationals
R : set of reals

Defining a mathematical set

- exhaustive enumeration: list everything

S = {1, 2, 17}

- set abstraction: define a set using set operations or a "set builder notation" like our list comprehensions

Z : set of integers
$Z+$: set of positive integers
$\mathbb{N}$: set of non-negative integers
Q : set of rationals
R : set of reals

The empty set, which contains no elements: {} or $\emptyset$

The universal set, U

Set operations (what can you do with sets S and T?)

- Informally . . .

- S or S^c: set complement
  - set of elements that are in U (the universal set) but not in S
- $S \cup T$: set union
  - set of elements that are in S or in T
- $S \cap T$: set intersection
  - set of elements that are in S and in T
- S-T: set difference
  - set of elements that are in S and not in T

Set operations in set notation

- S^C: set complement
  - set of elements that are in U (the universal set) but not in S
  - $S^C = \{x \in U : x \in S\}$
- $S \cup T$: set union
  - set of elements that are in S or in T
  - ST $\{x : x \in S \text{ or } x \in T\}$
- $S \cap T$: set intersection
  - set of elements that are in S and in T
  - ST $\{x : x \in S \text{ and } x \in T\}$
- S-T: set difference
  - set of elements that are in S and not in T
  - ST $\{x : x \in S \text{ and } x \notin T\}$

$$U = \mathbb{Z}^+$$
$$A = \{n : n \geq 6\}$$
$$B = \{1, 2, 4, 5, 7, 8\}$$

$$A^C$$
$$A \cap B$$
$$A \cup B$$
$$|B|$$

What can you say about sets S and T?

- = : set equality
  - S and T contain the same elements
- ⊆ : subset
  - S contains T
- ⊂ : proper subset
  - S contains T and S does not equal T
- ⊇ : superset
  - T contains S
- ⊃ : proper superset
  - T contains S and T does not equal S

$$U = \mathbb{Z}^+$$
$$A = \{n : n \geq 6\}$$
$$B = \{1, 2, 4, 5, 7, 8\}$$

- Are either A or B a subset of the other?
- Given an example of a proper superset of B.

Functions

Definition 2.46: Function.
Let A and B be sets. A *function f from A to B*, written $f : A \to B$, assigns to each input value $a \in A$ a unique output value $b \in B$; the unique value b assigned to a is denoted by $f(a)$. We sometimes say that f *maps a to* $f(a)$.

- Example(s) of function(s) from {1,2,3} to {2,4,6}
- What is an example of a function

Defining functions

- symbolically
- exhaustively
- how would you define the function for "and"?
  - what does it map from/to?

Cartesian product

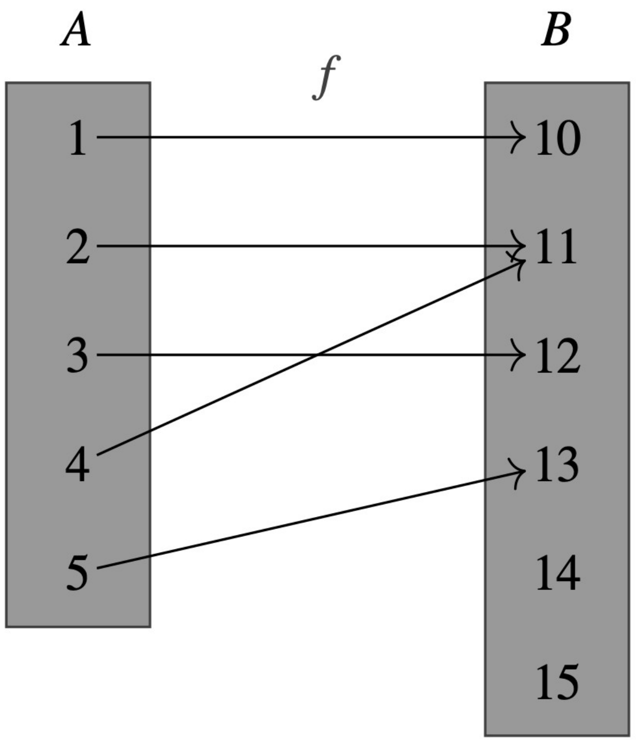
- The Cartesian product of two sets is written AxB and is defined as:

$$A \times B = \{ (x,y) : x \in A \text{ and } y \in B \}$$

- What is AxB if A = {1,2} and B = {true, false}?
- How would you define the function for "and"?
- How would you define a function which takes two real numbers and returns their average?

Definitions related to functions

- Given a function
  - the domain is the set A
  - the co-domain is the set B
  - the range (or the image) is the subset of B that are actually mapped to by an element in A.

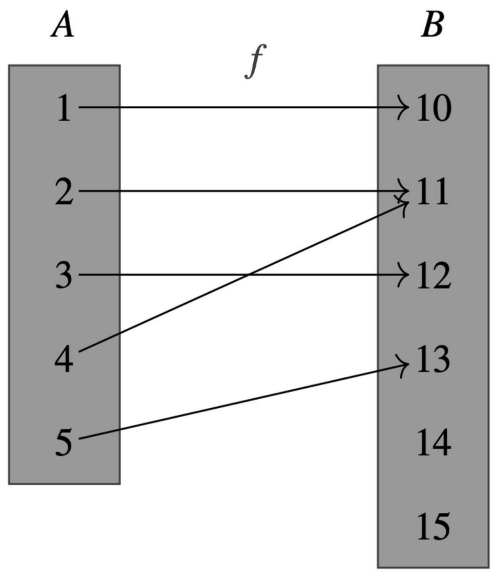


- Examples:
  - IsEven
  - Pow (haskell ^)
  - what's an example of a function whose domain, co-domain, and range are all the same?

classifying functions

- one-to-one: a function is one-to-one if, for every element of the co-domain, at most one element of the domain maps to it.

- onto: a function is onto if, for every element of the co-domain, there is an element of the domain that maps to it.
  - alternatively, a function is onto if the co-domain equals the range
- bijection: a function is a bijection if it is both one-to-one and onto



a little on Latex

```
\documentclass{article}
\usepackage{amsmath,amssymb}

\begin{document}
Hello world!

Let's define a few sets
\begin{itemize}
\item Here's a set with one element:
\[ S = \{x \in \mathbb{Z} : x+10=100 \}
\]
\item Here's a set with an infinite
number of elements: $S^C$
\end{itemize}

\end{document}
```

Hello world!
Let's define a few sets

- Here's a set with one element:

$$S = \{x \in \mathbb{Z} : x + 10 = 100\}$$

- Here's a set with an infinite number of elements: S^C

- note: environments, math mode, packages
- a quick intro:
  https://www.overleaf.com/learn/latex/Learn_LaTeX_in_30_minu