Mathematics and Problem Solving

Lecture 8

Set Theory 2

"To choose one sock from each of infinitely many pairs of socks requires the Axiom of Choice, but for shoes the Axiom is not needed."

Bertrand Russel

Overview

- Operators
  - Cardinality
  - Power sets
  - Compliments
  - Generalised operators
  - Cartesian Products
- Typed Set Theory
- Set Comprehensions

Cardinality

Cardinality

- The **cardinality** of a set is the number of elements it contains
  - It is the size of the set
- The unary operator # gives the cardinality of a set, e.g.
  - # { a } = 1
  - # { a, b } = 2
  - # { a, b, c } = 3

Cardinality

Law 8.1:

The cardinality of the empty set is 0

$$\# \emptyset = 0$$

Cardinality

Exercise 1:

The union operator increases cardinality whereas intersection and difference restrict it.

Show the following are true for the sets $S = \{ a, b, c, d \}$ and $T = \{ a, b, c, x, y z \}$.

1. $\#(S \cap T) = \#S - \#(S \setminus T)$
2. $\#(S \cup T) = \#S + \#T - \#(S \cap T)$

Cardinality of Intersection

Law 8.2:

The cardinality of the union of *S* and *T* is equal to the cardinality of S minus the cardinality of the intersection of *S* and *T*:

$$\#(S \cap T) = \#S - \#(S \setminus T)$$

Cardinality of Union

Law 8.3:

The cardinality of the union of *S* and *T* is equal to the cardinality of S plus the cardinality of T minus the cardinality of the intersection of S and T

$$\#(S \cup T) = \#S + \#T - \#(S \cap T)$$

Cardinality of Difference

Law 8.4:

The cardinality of the intersection of S and T is equal to the cardinality of S minus the cardinality of the intersection of S and T:

$$\#(S \setminus T) = \#S - \#(S \cap T)$$

Summary of Cardinality

- The cardinality operator # gives the number of elements in a set
- Cardinality Laws
  - Law 8.1: $\# \emptyset = 0$
  - Law 8.2: $\#(S \cap T) = \#S - \#(S \setminus T)$
  - Law 8.3: $\#(S \cup T) = \#S + \#T - \#(S \cap T)$
  - Law 8.4: $\#(S \setminus T) = \#S - \#(S \cap T)$

Cardinality

Finite and Infinite Sets

Finite and Infinite sets

- With a finite set, the process of counting the elements would eventually stop
  - So far all the sets we've worked with have been finite sets
- This is not the case for infinite sets, e.g.
  - $\mathbb{N}$ – the set of all natural numbers
  - $\mathbb{Z}$ – the set of all integers
  - $\mathbb{R}$ – the set of all real numbers

Infinite and finite sets

Exercise 2:

Which of the following are infinite and which of the following are finite sets:

1. All the grains of sand on all the beaches on earth
2. The set of all even integers
3. The empty set
4. The set of all Real numbers minus all the Natural numbers

Power Sets

Power sets

- The **power set** of a set *S* is the **set of all subsets** of *S*
  - Remember *S* and ∅ are subsets of *S*
- The unary operator $\mathbb{P}$ returns the power set
  - $\mathbb{P}(S)$ is the power set of *S*

$$S = \{1, 2, 3\}$$

$$\mathbb{P}(S) = \{\emptyset, \{1\}, \{2\}, \{3\}, \{1, 2\}, \{2, 3\}, \{1, 3\}, \{1, 2, 3\}\}$$

Power sets

Exercise 3:

For the given sets $S = \{ a, b, c \}$ and $T = \{ 1, 2 \}$ calculate the following:

1. $\mathbb{P}(S)$
2. $\mathbb{P}(T)$
3. $\mathbb{P}(S \cap T)$
4. $\mathbb{P}(\emptyset)$
5. $\mathbb{P}(\{\emptyset\})$

Definitition of Power Set

Law 9.1:

The set S is an element of the Power set of T if and only if S is a subset of T:

$$S \in \mathbb{P}(T) \iff S \subseteq T$$

Power Set and the Empty Set

Law 9.2:

The empty set is an element of the power set of every set

$$\emptyset \in \mathbb{P}(S)$$

Cardinality of Power sets

Law 9.3:

The cardinality of the power set of a given set S, is equal to two to the power of the cardinality of S. Thus:

$$\#(\mathbb{P}(S)) = 2^{\#(S)}$$

Power sets

Exercise 5:

Calculate the following for set $S = \{ a, b \}$ and set $T = \{ \varnothing, \{ 1 \}, \{ 2 \} \}$

1. $\mathbb{P}(T)$
2. $\mathbb{P}\,\mathbb{P}(S)$

Power sets and Intersection

Law 9.4:

For a given set *R* which is an element of the power set of *S*, it is always the case that the intersection of *R* and *S* is equal to *R*.

$$R \cap S = R, \text{ where } R \in \mathbb{P}(S)$$

Summary of Power Set Laws

- The power set operator $\mathbb{P}$ gives the set of all subsets
- Power Set Laws
  - Law 9.1: $S \in \mathbb{P}(T) \Longleftrightarrow S \subseteq T$
  - Law 9.2: $\varnothing \in \mathbb{P}(S)$
  - Law 9.3: $\#(\mathbb{P}(S)) = 2^{\#(S)}$
  - Law 9.4: $R \in \mathbb{P}(S) \Longleftrightarrow R \cap S = R$

Compliments

Absolute Compliment

- The **complement** of a set is everything that is not in that set, relative to some domain
- Some versions of set theory include a Universal set U
  - In such systems, we can calculate the absolute complement of a set with respect to U, notated:

$$A^C = U \setminus A$$

- The existance of a Universal set can lead to paradox, so in most popular set theories it is not used

Compliment relative to Power Set

- Instead of using a Universal set, we can describe the complement of a set relative to a power set that contains it. Thus for sets *S* and *T* defined:

$$S = \{ a, b, c \}$$

$$T = \{ a, c \}$$

- Where *S* is an element of the power set of *S* the compliment of *T*, denoted T^{-}, is equal to $S \setminus T$.

Compliments

Exercise 7:

For sets $S = \{ a, b, c \}$ and $T = \{ a, c \}$, wherein $T \in \mathbb{P}(S)$, calculate:

1. $\mathbb{P}(S)$
2. $S \setminus T$
3. T^{-}

Compliments

Law 10.1:

For a given set *T*, which is an element of the power set of *S*, the compliment of T is equal to the set difference in *S* and *T*

$$T^{-} = S \setminus T, \qquad \text{where } T \in \mathbb{P}(S)$$

Compliments

Law 10.2:

For a given set *T*, which is an element of the power set of S, The union of *T* and its compliment is equal to the *S*

$$T \cup T^{-} = S, \quad \text{where } T \in \mathbb{P}(S)$$

Compliments

Law 10.3:

For a given set *T*, which is an element of the power set of S, The intersection of *T* and its compliment is equal to the Ø.

$$T \cap (T^{-}) = \varnothing, \qquad \text{where } T \in \mathbb{P} (S)$$

Compliments

Exercise 8:

For the set $T = \{1, 2, 3, 4, 5\}$, calculate the following:

1. $\{1, 3, 5\}^-$
2. $\varnothing^-$
3. $\{1, 2, 3, 4, 5\}^-$

Summary of Complements

- A complement is everything not in a set, relative to a domain
- Laws of Complements
  - Law 10.1: $T^{-} = S \setminus T, \quad \text{where } T \in \mathbb{P}(S)$
  - Law 10.2: $T \cup T^{-} = S, \quad \text{where } T \in \mathbb{P}(S)$
  - Law 10.2: $T \cap (T^{-}) = \varnothing, \quad \text{where } T \in \mathbb{P}(S)$

Generalised Operations

Verbosity of Binary Operators

- The syntax for operators introduced so far can be verbose
  - For example, to describe the union of the sets *A*, *B*, *C*, *D*, *E*, we write

$$X = \{ A \cup B \cup C \cup D \cup E \}$$

- A solution to this is to use **generalised operators**

Generalised Union

- Generalised Union is written with a large $\bigcup$, and then all of the sets to union are given, for example

$$X = \bigcup \{ A, B, C, D, E \}$$

is equivalent to

$$X = \{ A \cup B \cup C \cup D \cup E \}$$

Generalised Union

Law 11.1:

Given a set of sets A and an element a,

a is an element of $\bigcup A$ iff there exists a set in A that contains a

$$a \in \bigcup A \iff \{ S \in A \mid a \in S \}$$

Generalised Intersection

- Generalised Union is written with a large $\bigcap$, and then all of the sets to intersect are given, for example

$$\bigcap \{ A, B, C, D, E \}$$

  is equivalent to

$$X = \{ A \cap B \cap C \cap D \cap E \}$$

Generalised Operations

Law 11.2:

Given a set of sets A and an element a,

a is an element of $\bigcap$ A iff a is an element every set A

$$\mathrm{a} \in \bigcap A \iff \{\, \forall S \in A \mid a \in S \,\}$$

Cartesian Product

Introducing Cartesian Product

- Cartesian Product is another binary operator
  - Uses the symbol ×, e.g.
  - $R \times S$
- Generates a set of all possible of pairs (tuples) of elements of R and S

- For the given sets:

  $R = \{ a, b \} \quad S = \{1, 2\}$

- The cartesian product is:

  $R \times S = \{ (a, 1), (a, 2), (b, 1), (b, 2) \}$

- Note that order matters:

  $S \times R = \{ (1, a), (1, b), (2, a), (2, b) \}$

Sets of different size

- Sets do not have to be the same size to calculate Cartesian Products

$$R = \{ a, b, c \} \qquad S = \{ 1, 2 \}$$

$$\begin{aligned} R \times S = \{ & (a, 1), (a, 2), \\ & (b, 1), (b, 2), \\ & (c, 1), (c, 2) \} \end{aligned}$$

Exercise 14:

Give extensions for the following sets:

1. $\{ a \} \times \{ 1, 2, 3 \}$
2. $\{ 1, 2 \} \times \{ 1, 2 \}$
3. $\{ a, b, c \} \times \emptyset$

Exercise 15:

Given the sets:

$$\text{People} = \{ \text{Alice, Bob, Carla} \}$$

$$\text{Cars} = \{ \text{Skoda, Fiat, BMW} \}$$

Give extensions for the following:

1. $\text{People} \times \text{Cars}$
2. $\mathbb{P}\,(\text{Cars} \times \text{People})$

Cardinality of Cartesian Products

Law 12.1:

The size of a Cartesian product is determined by the cardinality of the elements of the considered sets. Thus:

$$\#(S \times R) = \#X \times \#Y$$

Exercise 16:

Using Law 12.1, determine the number of elements in the following Cartesian products:

1. $\{ a, b, c \} \times \{ 1, 2, 3 \}$
2. $\{ a, b \} \times \{ 1, 2, 3 \}$
3. $\varnothing \times \{ 1, 2, 3 \}$

- Hint
  - Law 12.1: $\#(S \times R) = \#X \times \#Y$

Functions

Functions in Set Theory

- Functions in set theory can be defined as pairs
- $A \to B$
- $A \times B$

Types

Typed Set Theory

- In our strictly typed set theory we cannot put elements together that are not of the same type
  - This makes Typed Set Theory (and Type Theory especially) highly suited to computer science, e,g, programming
- Moreover, we can't make sets of just anything

Cantor’s Paradox

- Imagine we had a set S that contained every set.

$$S = \{ X \mid X \text{ is a set } \}$$

- Every subset of S is a set, so it must also be an element of S. Therefore

$$\mathbb{P}(S) \subseteq S$$

- However (according to law 9.3) the power set of a given set is always greater than the cardinality of the set itself, so

  the cardinality of S must be bigger than itself!

Russel’s Paradox

- Imagine we had a set S that contained all sets that are not elements of themselves

$$S = \{ X \mid X \text{ is not an element of itself} \}$$

- Is $S \in S$?
  - If no, then it should be, because “S is not an element of itself”
  - If yes, then it shouldn’t be, because it’s not the case that “S is not an element of itself”

Typed set theory

- Set theories need a way to avoid paradoxes of this sort
  - Needs to limit what sorts of sets are allowed
- A typed set theory says that all sets must have a type, and can only contain elements of that type

Typed set theory

- Typed set theory is not only about ensuring that elements are of the same type. It's also a requirement that all sets must have a type.
- It therefore follows that operators can only be applied to sets of the same type.

Typed set theory

Exercise 9:

Which of the following are legal operations within typed set theory and which are not:

1. $\{1, 2, 3\} \cap \emptyset$
2. $\{1, 2, 3\} \cap \{0, 1, 2\}$
3. $\{a, b\} \setminus \{\emptyset, \{\emptyset\}\}$
4. $\{\{1\}\} \in \{1, 2, 3\}$
5. $\{1\} \in \{\{1\}, \{2\}, \{3\}\}$
6. $1 \in \{1, 2, 3\}$

Empty Set in Typed Set Theory

- Consider the following:
  - For the given set, this is a legal (and true) statement: $\varnothing \subseteq \{1, 2, 3, 4\}$
  - Because of this we can also perform the following operation

    $(\varnothing \cap \{ 1, 2, 3, 4 \}) = \varnothing$
- With this consider the equally legal statement: $\varnothing \cap \{a, b, c, d\}$ which also results in $\varnothing$. Taking both these results can we then say: $\varnothing \Longleftrightarrow \varnothing$?
- The answer is no because they are different types of empty set. As such, if this was expressed in an equation of typed set theory we would consider it illegal.

Set Comprehension

Basic Set Builder Notation

- The way we notate sets is called **set builder notation**
- We can use it define sets **extensionally**, as we have seen
  - { a, b, c }
- Or we can define sets **intentionally**
  - { x | x is a cat }

Intentional Definitions

- A simple intensional definition has three parts, for example in

$$\{ x \mid \Phi(x) \}$$

the parts are

  - x – a variable
  - | - a vertical bar, often read "such that"
  - $\Phi(x)$ – some predicate, or property, that must hold of x for it to be a member of the set

Basic Intentional Definitions

Exercise n:

Describe in words the contents of the following sets:

1. $\{ x \mid x \text{ is a cat}\}$
2. $\{ x \mid x > 4 \}$
3. $\{ x \mid x \text{ is tall and } x \text{ is a giraffe} \}$
4. $\{ (x, y) \mid x \text{ is tall and } y \text{ is a giraffe} \}$

Specifying a Domain

- In these examples, we did not specify a domain, as it was clear from context
- However, what is the extension of the set $\{ x \mid x \text{ is odd} \}$
  - Odd numbers? Odd people?
- Let's make this clear, by specifying the domain, or type of x
  - $\{ x : \mathbb{N} \mid x \text{ is odd} \}$
  - This is clear, the set contains only odd numbers (specifically odd natural numbers)

Set Comprehension Template

- In general, set comprehensions follow the template:

  { declaration | predicate }

- The **declaration** enumerates entities of required types
  - Previously we had "$x : \mathbb{N}$", a variable called x of type $\mathbb{N}$
  - Another notation for this is "$x \in \mathbb{N}$"
- The **predicate** filters them, keeping all entities that satisfy the predicates criteria.
  - Previously we used "*x* is odd"

Basic Set Comprehension Examples

- More examples of this
  - Even_Naturals = $\{ n : \mathbb{N} \mid n \bmod 2 = 0 \}$
  - Car_Owners = { p : People | p owns a car }
  - Tall_Giraffes = { g : Giraffes | g is tall }
- These all follow the pattern

{ variable : Type | predicate }

Exercise 10:

Define by extension the following sets:

1. $\{ n : \mathbb{N} \mid n > 0 \wedge n < 10 \}$
2. $\{ n : \mathbb{N} \mid n < 100 \wedge n \bmod 10 = 0 \}$
3. $\{ n : \mathbb{R} \mid (n \times 2 = 14) \vee (3 + n = 10) \}$

Set Comprehension

Exercise 11:

Define the following by set comprehension

1. The set of all taxis
2. The set of all home owners
3. The set of all natural numbers

Set Comprehension

Introducing Terms

Complex Variables

- Earlier you saw the example

$$\{ (x, y) \mid x \text{ is tall and } y \text{ is a giraffe} \}$$

- This notation describes a set of tuples (x, y), where
  - the first of the pair is a tall thing, and
  - the second of the pair is a giraffe
- But this gets uglier when defining types for our variables. Should we write…

$$\{ (x\text{: Object}, y\text{: Object}) \mid x \text{ is tall and } y \text{ is a giraffe} \} \quad ?$$

$$\{ (x, y) : (\text{Object}, \text{Object}) \mid x \text{ is tall and } y \text{ is a giraffe} \} \quad ?$$

Introducing Terms

- There is a clearer notation to solve this.
- As well as having a declaration and a predicate, we can add a 'term', following the template

$$\{\ \text{declaration} \mid \text{predicate} \bullet \text{term}\ \}$$

- The term describes what will go into the generated set.

$$\text{Owner\_Address} = \{\ p : \text{People} \mid p \text{ owns a house} \bullet \text{address}(p)\}$$

Parts of a Set Comprehension

- So the parts of the template below are as follows

$$\{ \text{declaration} \mid \text{predicate} \bullet \text{term} \}$$

  - The **declaration** enumerates entities of the desired types
  - The **predicate** filters these
  - The **term** describes what is added to the set
- We could see this instead as

$$\{ \text{generate} \mid \text{filter} \bullet \text{Action} \}$$

Using Terms

Exercise 12:

Define by comprehension (using definition, predicate and term) sets that generate the following:

1. Students in a class who are vegan
2. The age of all male staff
3. People who use android phones

- Hint:
  - Remember the template

{ declaration | predicate • term}

Set Comprehension

- Declaration, predicate and term may be combined as follows:

  { declaration | predicate }

  { declaration | predicate • term }

  { declaration • term }

- E.g. to generate a set of car registrations, we can do:
  - { *r* : registration | r is a car registration }
    - But it's awkward
  - { *c* : car | true • registration }
    - But the predicate is unnecessary
  - { *c* : car • registration }
    - Is clearest in this case

Set Comprehension

Exercise 13:

List by extension the first four elements of the following sets:

1. $A = \{ n : \mathbb{N} \bullet n + n \}$
2. B = $\{ n : \mathbb{N} \bullet n \bmod 5 \}$
3. $C = \{ n : \mathbb{N} \bullet n^2 \}$
4. $D = \{ n : \mathbb{N} \bullet n \}$

Set Comprehensions

Returning to Cartesian Product

Defining Cartesian Product

- We can define cartesian product by set comprehension

$$R \times S = \{ r : R; s : S \bullet (r, s) \}$$

- This defines cartesian product as

  the set of all ordered pairs (r, s)

  such that $r \in R$ and $s \in S$

More Set Comprehensions

Characteristic Tuple

- Cartesian products introduced us to the characteristic tuple. That it, the tuple that defines a given set comprehension. For the given set comprehension:

$$\{ n : \mathbb{N} \mid n < 5 \bullet n \bmod 2 \}$$

- n is the characteristic tuple.
- Whenever a set comprehension contains two or more elements and no term, it will still return a cartesian product

Multiple Characteristic Tuples

- In a situation where two variables are present in the way illustrated below, we have two characteristic tuples, which will return an ordered pair (p, c).

$$\{ \; p : \text{People}; \; c : \text{Cars} \mid \mathit{true} \bullet (p, c) \; \}$$

- In the above the term part is unnecessary. Thus if you encountered this the return values would be ordered pairs in the order in which the variables are declared in the declaration.

Cartesian Products

- Cartesian products can work on any number of sets at a time to produce tuples of any finite size. For example,

$$\text{Members} = \{\, x : \text{Name} \times \text{Age} \times \text{Address} \mid \textit{true} \,\}$$

Extracting Elements from Cartesian Products

- Further, it is also possible to extract elements from a Cartesian product as follows:

$$Member\_Ages = \{ x : Name \times Age \times Address \mid true \bullet x.2 \}$$

- In the above example the dot operator combined with the number selects the element from the Cartesian product that we are concerned with.

Cartesian Products

Exercise 16:

Assuming the following database table, give set comprehensions to return the following:

1. Everyone in Gryffindor
2. The Student id for all students under 16
3. All possible pairings for cross house study buddies

ID	Name	Age	House
12345	Harry	15	Gryffindor
12346	Ron	15	Gryffindor
12347	Hermione	16	Gryffindor
12348	Draco	16	Slytherin
12349	Crabbe	15	Slytherin
12350	Luna	13	Ravenclaw