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.
  - $\# \{ a \} = 1$
  - $\# \{ a, b \} = 2$
  - $\# \{ a, b, c \} = 3$

Exercise 1:

What is the cardinality of the following sets?

1. { a, b, c }
2. The set of letters in the alphabet
3. The empty set
4. The set of subsets of {a, b, c}

- Hint:
  - The cardinality of a set is the number of elements it contains

Cardinality

Law 8.1:

The cardinality of the empty set is 0

$$\#\,\varnothing = 0$$

Exercise 2:

What is the cardinailities of each set when S = {a, b, c} and T = {b, c, d}?

Write a formula for each set for any sets S, T.

1. $S \cup T$
2. $S \cap T$
3. $S \setminus T$

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:

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

Cardinality

Exercise 3:

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

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

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 4:

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

Summary of Cardinality

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

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*

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

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

Power sets

Exercise 5:

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. $\mathbb{P}(\varnothing)$
5. $\mathbb{P}(\{\varnothing\})$

- Hint:
  - $\mathbb{P}(\mathrm{S})$ is the power set of S
  - The power set of a set S is the set of all subsets of S
    - Remember S and $\varnothing$ are subsets of S

Defininitition 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

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

Power Set Laws

Law 9.3:

For any given set S, S is an element of the power set of itself

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

Cardinality of Power sets

Law 9.4:

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 6:

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

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

Power sets and Intersection

Law 9.5:

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

Relative Compliment

- We can describe the complement of a set relative to its superset
  - We will denote this complement T^{-}, relative to its superset S
    - where $T \in \mathbb{P}(S)$
    - Or, equivalently, $T \subseteq S$
- For example, for the sets S and T below, where $T \in \mathbb{P}(S)$

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

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

$$T^{-} = \{ c \} = S \setminus T$$

Compliments

Exercise 7:

What is the relative complement of the following sets, with respect to S = {a, b, c, d}

1. { a }
2. {a, c, d}
3. Ø
4. {a, b, c, d}

- Hint:
  - Where *S* is a subset of *S* the compliment of *T* relative to S, denoted T^{-}, is equal to $S \setminus T$.

Definition of Compliment

Law 10.1:

For a given set $T \subseteq S$, the compliment of T is equal to the set difference in S and T

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

Union of Compliments

Law 10.2:

For a given set $T \subseteq S$, the union of T and its compliment is equal to the S

$$T \cup \bar{T} = S \qquad \text{where } T \in \mathbb{P}(S)$$

Intersection of Compliments

Law 10.3:

For a given set $T \subseteq S$, the intersection of T and its compliment is equal to the empty set.

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

Exercise 8:

What are the following:

1. $T \cup (T^{-} \cap T)$
2. $T \cap (T \cup T^{-})$
3. $T \cap (T^{-} \cap T)$

- Hint:
  - Law 10.1: $T^{-} = S \setminus T$, where $T \in \mathbb{P}(S)$
  - Law 10.2: $T \cup T^{-} = S$, where $T \in \mathbb{P}(S)$
  - Law 10.3: $T \cap T^{-} = \emptyset$, where $T \in \mathbb{P}(S)$

Summary of Complements

- A complement is everything not in a set, relative to a domain
- Laws of Complements
  - Law 10.1: $T^{-} = S \setminus T$, where $T \in \mathbb{P}(S)$
  - Law 10.2: $T \cup T^{-} = S$, where $T \in \mathbb{P}(S)$
  - Law 10.3: $T \cap T^{-} = \emptyset$, where $T \in \mathbb{P}(S)$

Quantifiers

Quantifiers

- This will be a brief introduction to quantifiers, sufficient to understand the following slides
- Quantifiers in predicate, or first-order logic, allow you to make true or false statements about the contents of sets
  - There are two quantifiers

$$\exists \quad \forall$$

  - Read these ‘there exists’ and ‘for all’

Quantified Statements of Logic

- Quantified statements of logic have the form

$$Q\, e \in S\, .\, P$$

- Q is a quantifier
- e is a variable bound by the quantifier
- S is a set
  - (which e is drawn from)
- Read ‘ . ’ as ‘such that’
- P is a proposition
  - (which often includes e as a variable)

Existential Quantification

- $\exists$ is the **existential** quantifier
  - It says there is at least one element satisfying the proposition that follows

$$\exists x \in \mathbb{N} . x > 3$$

'There exists some x (where x is a natural number), such that x > 3'

Universal Quantification

- $\forall$ is the **universal** quantifier
  - It says that all elements satisfy the proposition that follows

$$\forall x \in \mathbb{N}^{+} . x > 0$$

‘For all x (where x is a positive natural number), it is true that x > 0’

Nesting Quantifiers

- Quantifiers can be nested, allowing for complex expressions

$$\exists a \in \mathbb{N}, b \in \mathbb{N} \;.\; \forall c \in \mathbb{N} \;.\; a + b < c$$

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

$$a \in \bigcup A \iff \exists S \in A \,.\, a \in S$$

(Read the right hand side as ‘there exists an S in A such that a is an element of S’

Generalised Intersection

- Generalised Intersection 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

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

(Read the right hand side of this as ‘For all elements S in A, a is an element of S’)

Exercise 9:

Answer the following for the sets A = {a}, B = {a, b, c}, C = {a, c, d}

1. $\bigcap\{A, B, C\}$
2. $\bigcup\{A, B, C\}$

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 10:

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 11:

Given the sets:

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

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

Give extensions for the following:

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

Cardinality of Cartesian Products

Law 8.5:

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

$$\#(S \times R) = \#S \times \#T$$

Cartesian Products

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

$$a \times b \times c = \{ (a_1, b_1, c_1), (a_1, b_1, c_2), \dots \}$$

Exercise 12:

Using Law 8.5, 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. $\varnothing \times \{ 1, 2, 3 \}$
4. $\{ a, b \} \times \{ 1, 2 \} \times \{ \alpha, \beta \}$

- Hint
  - Law 8.5: $\#(S \times R) = \#S \times \#R$

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 9.4) the cardinality of 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.
  - We often write the type of a set following a colon, for example:

    $S : \mathbb{N}$ (S is a set of type $\mathbb{N}$ (natural numbers))

    T : Cats (T is a set of cats)

Typed set theory

Exercise 13:

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

1. $S \cap T$, where $S : \mathbb{N}$ and $T : Cats$
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: $\emptyset \subseteq \{1, 2, 3, 4\}$
  - Because of this we can also perform the following operation

    $(\emptyset \cap \{1, 2, 3, 4\}) = \emptyset$
- With this consider the equally legal statement: $\emptyset \cap \{a, b, c, d\}$ which also results in $\emptyset$. Taking both these results can we then say: $\emptyset \Longleftrightarrow \emptyset$?
- 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 14:

Describe in words the contents of the following sets:

1. { x | x is a cat}
2. { x | x > 4 }
3. { x | x is tall and x is a giraffe }
4. { (x, y) | x is tall and y 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 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 15:

Define by extension the following sets:

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

Set Comprehension

Exercise 16:

Define the following by set comprehension

1. The set of even numbers
2. The set of numbers greater than 10
3. The set of subsets of a set of numbers, S
4. The intersection of three sets of numbers, R, S, T

Set Comprehension

Introducing Terms

Complex Variables

- Earlier you saw the example

  { (x, y) | x is tall and y 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: Object, y: Object) | x is tall and y is a giraffe } ?

  { (x, y) : (Object, Object) | x is tall and y is a giraffe } ?

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

{ declaration | predicate ● term }

- The term describes what elements will go into the generated set, e.g.

{ x □Animal; y: Animal | x is tall and y is a giraffe ● (x, y)}

Parts of a Set Comprehension

- So the parts of the template below are as follows

{ declaration | predicate • 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

{ generate | filter • Action}

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 17:

List four elements in the following sets (for the lowest n)

1. $\{ n\colon \mathbb{N} \mid n > 3 \bullet n \}$
2. $\{ n\colon \mathbb{N} \bullet n + n \}$
3. $\{ n\colon \mathbb{N} \mid n \bmod 3 = 0 \bullet n \}$
4. $\{ n\colon \mathbb{N} \bullet n \bmod 5 \}$
5. $\{ n\colon \mathbb{N} \bullet n^2 \}$

Using Terms

Exercise 18:

Define the following sets by comprehension:

1. $\{3, 6, 9, 12, \ldots\}$
2. $\{100, 102, 104, 106, \ldots\}$
3. $\{1, 0.5, 0.25, 0.125, \ldots\}$
4. $\{4, 5, 6, 7\}$

- Hint:
  - Remember the template

{ declaration | predicate • term}

Set Comprehensions

Cartesian Products

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 $s \in S$
- Here (r, s) are **characteristic tuples** of this set comprehension

Characteristic Tuples

- The characteristic tuples are assumed if no term is specified
  - The following two expressions return the same set

$$\{ a: \mathbb{N}; b: \mathbb{N}; c: \mathbb{N} \mid a > b > c \bullet (a, b, c) \}$$

$$\{ a: \mathbb{N}; b: \mathbb{N}; c: \mathbb{N} \mid a > b > c \}$$

- If a set comprehension contains two or more elements (e.g. s :S; r: R) and no term it will return a cartesian product (e.g. a subset of S x R)

Multiple Characteristic Tuples

- Consider the example

$$\{\ p : \text{People};\ c : \text{Cars} \mid \mathit{true}\ \}$$

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

Question:

Are these two equivalent? Why not?

Exercise 19:

Give 4 elements from the following sets

1. $\{ a: \mathbb{N}; b: \mathbb{N}; c: \mathbb{N} \mid a > b > c \}$
2. $\{ a: \mathbb{N}; b: \mathbb{N}; c: \mathbb{N} \mid a < b < c \ \land a \neq b \neq c \bullet (a, c) \}$
3. $\{ a: \mathbb{N}; S: \mathbb{P}(\mathbb{N}); \mid a \in S \ \land \#(S) = 3 \}$

Cartesian Products

- One of our types in a set comprehension might be a cartesian product. For example,

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

Question:

How is this different to the following?

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

Extracting Elements from Cartesian Products

- We can extract elements from a Cartesian product, for example:

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

- In the term above, the dot operator ‘ . ’ selects the second element (age) out of the cartesian product

Extracting Elements from Cartesian Products

For example, if

- Name = {John, Mary}
- Age = {22, 19}
- Address = { 1 A St. , 2 B St. }

and

- Post_Address = { x : Name × Age × Address | true • (x.1, x.3)}

then

- Post_Address = { (John, 1 A St.), (Mary, 2 B St.) }

Cartesian Products

Exercise 20:

Assuming types ID, Name, Age, House with contents as shown, give set comprehensions for:

1. Everyone in Gryffindor
2. The ID of 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

Summary

Overview

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