

Computational Thinking Discrete Mathematics

Topic 03 — Collections

Number Theory

Logic

Lecture 01 — Sets

Dr Kieran Murphy 

Department of Computing and Mathematics,
SETU (Waterford).
(kieran.murphy@setu.ie)

Graphs and
Networks

Collections

Autumn Semester, 2025/26

Outline

- Definition of a set
- Relationships between sets
- Set operations

Outline

1. Defining Sets	2
1.1. Number Sets	5
1.2. Set Builder Notation	6
1.3. Venn Diagrams	10
1.4. Python Implementation	11
2. Relationships Between Sets	12
2.1. Subsets and Proper Subsets	14
2.2. Power Set	17
2.3. Python Implementation	19
3. Set Operations	20
3.1. Intersection, Union, Set Difference, ...	21
3.2. Cartesian product	27
3.3. Python Implementation	29
4. Proving Equivalence	31

Sets

Sets are fundamental discrete structures that form the basis of more complex discrete structures such as graphs, relational data bases, etc.

Definition 1 (Set)

A **set** is an unordered collection of distinct well-defined objects (called **elements**).

The **Set** definition is a perfect example of how all definitions need to be read.

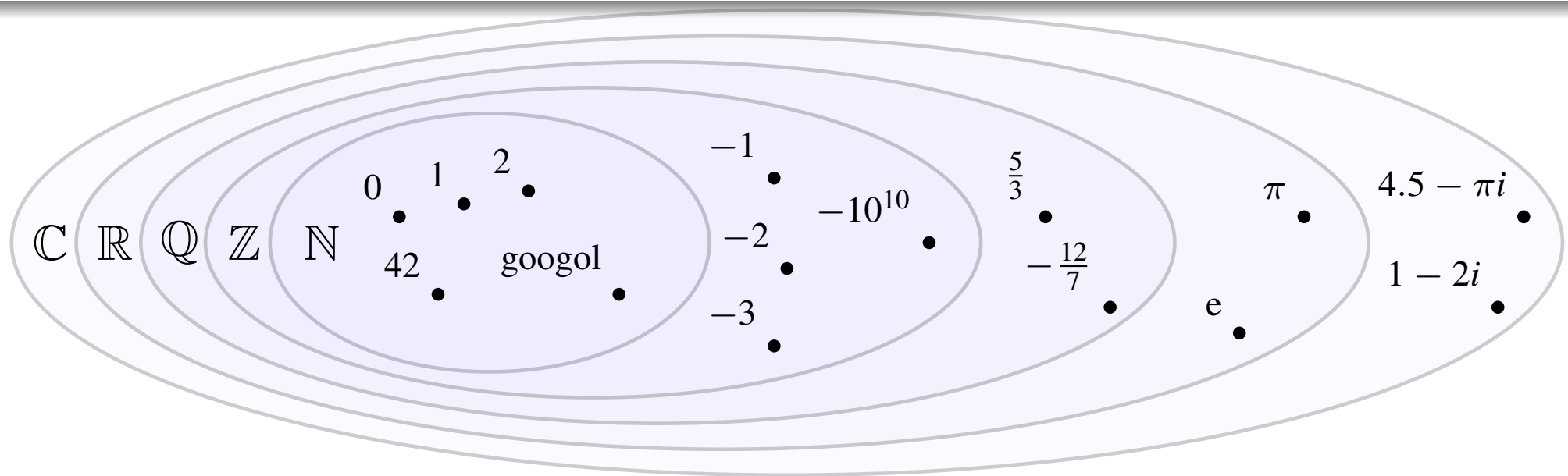
Take care to read and parse the definition of sets carefully:

- “**unordered**” means order is not important.
 - So two sets with the same elements but in different order are equal.
- “**collection**” means zero or more items.
- “**distinct**” means elements are unique.
 - So adding an element more than once has no effect.
- “**well-defined**” means that we have a clear rule for deciding what is in the set and what is not in the set.
 - So the “set of all healthy foods” is not a set.

- “ x is not an element of A ”

4 of 34

Number Sets



- $\mathbb{N} = \{0, 1, 2, 3, 4, 5, \dots\}$ (Natural Numbers)
 - Contains zero and the positive integers.
- $\mathbb{Z} = \{\dots, -3, -2, -1, 0, 1, 2, 3, \dots\}$ (Integers)
- \mathbb{Q} (Rational Numbers)
 - Any number that can be expressed as a fraction.
- \mathbb{R} (Real Numbers)
 - Rational and irrational numbers
- \mathbb{C} (Complex)

Set Builder Notation

Another way of describing sets is to use **set builder notation**.
For example, we could define the set of rational numbers as

$$\mathbb{Q} = \{a/b \mid a, b \in \mathbb{Z}, b \neq 0\}$$

where

- a/b indicates that a typical element of the set is a “fraction.”
- The vertical line, “ \mid ”, is read as “such that” or “where”.
Note: Many authors use a colon, “ $:$ ” instead of the vertical line “ \mid ”.
- $a, b \in \mathbb{Z}$ is an abbreviated way of saying a and b are integers.
- Commas are usually read as “and.”

The above mathematical statement can be read as

\mathbb{Q} is the set of things that can be expressed as a/b where a and b are integers and $b \neq 0$.

Question 1:

List any four elements of each of the following sets:

- a) $\{k \in \mathbb{N} \mid k - 1 \text{ is a multiple of } 7\}$
- b) $\{x \mid x \text{ is a fruit and its skin is normally eaten}\}$
- c) $\{x \in \mathbb{Q} \mid \frac{1}{x} \in \mathbb{Z}\}$
- d) $\{2n \mid n \in \mathbb{Z}, n < 0\}$
- e) $\{s \mid s = 1 + 2 + \cdots + n, n \in \mathbb{N}, n \geq 1\}$

Question 2:

List all elements of the following sets:

- a) $\{\frac{1}{n} \mid n \in \{3, 4, 5, 6\}\}$
- b) $\{\alpha \in \text{the alphabet} \mid \alpha \text{ precedes F}\}$
- c) $\{-k \mid k \in \mathbb{N}\}$
- d) $\{n^2 \mid n = -2, -1, 0, 1, 2\}$
- e) $\{n \in \mathbb{P} \mid n \text{ is a factor of } 24\}$

Question 3:

Describe the following sets using set-builder notation.

- a) $\{5, 7, 9, \dots, 77, 79\}$
- b) the rational numbers that are strictly between -1 and 1
- c) the even integers
- d) $\{-18, -9, 0, 9, 18, 27, \dots\}$

Cardinality

Definition 2 (Cardinality)

Let A be a finite set. The number of different elements in A is called its **cardinality** and is denoted by $|A|$.

If $|A|$ is finite then A is said to be a **finite set**, otherwise it is an **infinite set**.

- The empty set, \emptyset , has cardinality zero, i.e.,

$$|\emptyset| = 0$$

- A **singleton set** is a set that has only one element.
- Note the difference between $\{a\}$ and a . The braces indicate that the object is a set, while a without the brace is an element.
- This difference also applies to the empty set, in that*

$$\emptyset \neq \{\emptyset\}$$

*If this is confusing, think of a bag containing a empty bag. Is the first bag empty?

Examples (Membership and Cardinality)

- Let $A = \{1, \{2\}, \{\{3\}\}\}$. Then

$$1 \in A$$

$$2 \notin A$$

$$\{2\} \in A$$

$$3 \notin A$$

$$\{3\} \notin A$$

$$\{\{3\}\} \in A$$

- Let $A = \{23, 24, \dots, 37, 38\}$.
Then $|A| = 38 - 23 + 1 = 16$

Note the “+1”.

- Let $B = \{1, \{2, 3, 4\}, \emptyset\}$.
Then $|B| = 3$

B has three elements, the number 1, the set $\{2, 3, 4\}$, the empty set \emptyset .

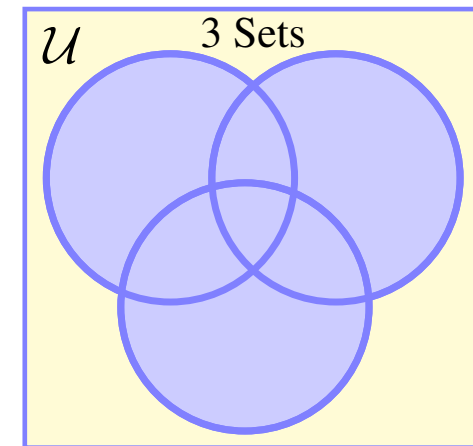
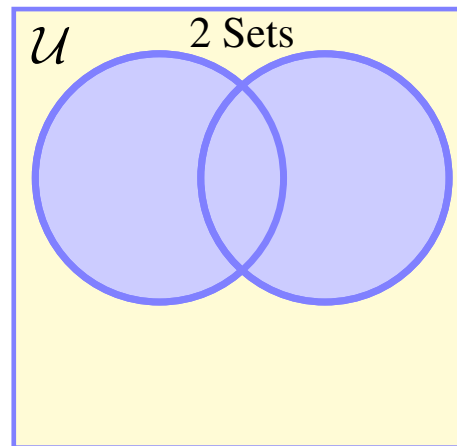
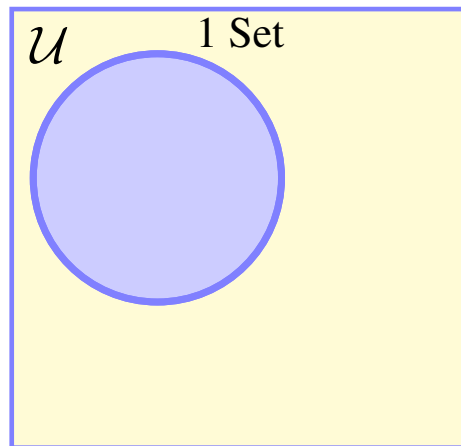
- Let $C = \{n \mid n < 100, n \text{ is prime}\}$.
Then $|C| = 25$

List and count the primes $\{2, 3, 5, 7, \dots, 97\}$.

Venn Diagrams

A **Venn diagram** is a graphical representation of sets that is effective when dealing with the relationship between a few sets. We use

- (overlapping) ovals to represent the individual sets.
- a rectangle to represent the **universal** set — a set
- an element is placed in exactly one region, based on which set, if any, it is a member of,



Python Implementation

`defining_sets .py`

```
1  A = set()
2  print(A)
3
4  A.add(3)
5  print(A)
6
7  A.add(3)
8  print(A)
9
10 A.add(2)
11 print(A)
12
13 A.add("Hello")
14 print(A)
15
16 A.add("All")
17 print(A)
18
19 B = {"This", "is", "a", "set"}
20 print(B)
```

- Create an empty set using `set()`
- Note: braces `{}` are used to represent a dictionary.
- Add individual items using `add`
- Items are unique so repeated add has no effect.
- Sets can contain 'anything'. Need quotes around strings
- Note order is not important!
- Can also define a set with elements

```
1  set()
2  {3}
3  {3}
4  {2, 3}
5  {2, 3, 'Hello'}
6  {2, 3, 'Hello', 'All'}
7  {'This', 'set', 'a', 'is'}
```

Outline

1. Defining Sets	2
1.1. Number Sets	5
1.2. Set Builder Notation	6
1.3. Venn Diagrams	10
1.4. Python Implementation	11
2. Relationships Between Sets	12
2.1. Subsets and Proper Subsets	14
2.2. Power Set	17
2.3. Python Implementation	19
3. Set Operations	20
3.1. Intersection, Union, Set Difference, ...	21
3.2. Cartesian product	27
3.3. Python Implementation	29
4. Proving Equivalence	31

Equal Sets

Definition 3 (Equal Sets)

Two sets A , and B are **equal** iff they contain the same elements.

Expressing this in predicate logic terms we have

$$\forall x \left[\underbrace{x \in A \leftrightarrow x \in B}_{x \text{ is in } A \text{ iff } x \text{ is in } B} \right]$$

which in terms of the IFTHEN operator is

$$\forall x \left[\underbrace{(x \in A \rightarrow x \in B)}_{\text{if } x \text{ is in } A \text{ then } x \text{ is in } B} \wedge \underbrace{(x \in B \rightarrow x \in A)}_{\text{if } x \text{ is in } B \text{ then } x \text{ is in } A} \right]$$

Example

The sets

- $A = \{1, 3, 5, 7, \dots\}$
- $B = \{n \mid n \in \mathbb{N}, \exists k(n = 2k + 1, k \in \mathbb{N})\}$
- $C = \{n \mid n \in \mathbb{N}, \text{remainder of } n \div 6 \in \{1, 3, 5\}\}$

are equal, despite the apparent difference in their definitions.

Subsets and Proper Subsets

Definition 4 (Subset)

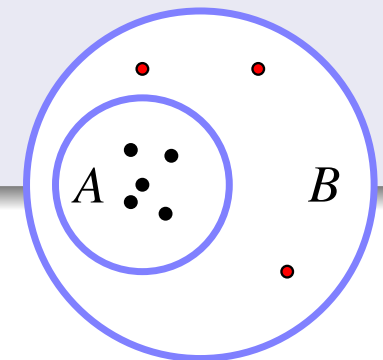
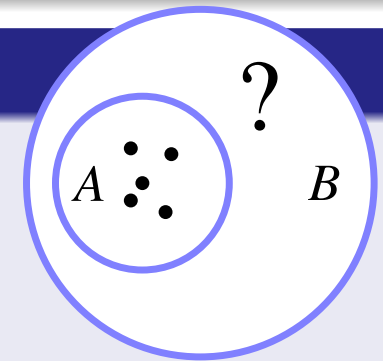
Set A is said to be a **subset** of B and we write

$$A \subseteq B$$

if and only if every element of A is also an element of B .

If, in addition, B contains **at least one** element not in A we say that A is a **proper subset** of B , and write

$$A \subset B$$



- In terms of predicate logic we have

- A is a subset of B

$$A \subseteq B \iff \forall x [x \in A \rightarrow x \in B]$$

- A is a proper subset of B

$$A \subset B \iff \forall x [x \in A \rightarrow x \in B] \wedge \exists x [x \in B, x \notin A]$$

- Note that the operators \subset and \subseteq play a similar role to $<$ and \leq .

The Empty Set is a Subset of Every Set

Example

Let $A = \{1, 2, 3, 4, 5, 6\}$, $B = \{2, 4, 6\}$, $C = \{1, 2, 3\}$ and $D = \{7, 8, 9\}$. Determine which of the following are true, false, or meaningless.

- a) $A \subset B$.
- b) $B \subset A$.
- c) $B \in C$.
- d) $C \in A$.
- e) $\emptyset \in A$.
- f) $\emptyset \subset A$.
- g) $A < D$.
- h) $3 \in C$.
- i) $3 \subset C$.
- j) $\{3\} \subset C$.

Power Set

If you collect all the subsets of set S into a new set, we get a set of sets ...

Definition 5 (Power Set)

The **power set** of a set S , denoted by $\mathcal{P}(S)$, is the set of all subsets of S .

The power set is a fundamental combinatorial object useful when considering all possible combinations of elements of a set.

Theorem 6 (Size of the power set)

Let S be a set such that $|S| = n$, then

$$|\mathcal{P}(S)| = 2^n$$

Example

Example 7

Let $A = \{a, b, c\}$, then the power set is

$$\mathcal{P}(A) = \{\emptyset, \{a\}, \{b\}, \{c\}, \{a, b\}, \{a, c\}, \{b, c\}, \{a, b, c\}\}$$

- Note that the empty set and the set itself are elements of the power set.
 - The empty set is a subset of every set ...

$$\emptyset \subseteq A$$

- Every set is a subset of itself ...

$$A \subseteq A$$

- Note that while $b \in A$, it is wrong to say $b \in \mathcal{P}(A)$. $b \neq \{b\}$
- However $\{b\} \in \mathcal{P}(A)$ since $\{b\} \subseteq A$.
- $\mathcal{P}(\emptyset) = \{\emptyset\}$

Python Implementation

```
1 A = {4,2}
2 B = {1,2,3,4,5}
3 print("A =", A)
4 print("B =", B)
5
6 print("Is A a subset of B?", A.issubset(B))
7 print("Is B a subset of A?", B.issubset(A))
8
9 print("Is A a superset of B?", A.issuperset(B))
10 print("Is B a superset of A?", B.issuperset(A))
11
12 print(A.issubset(A))
13
14 A = set()
15 print(A.issubset(A))
```

- Create two sets, A and B
- Test for subset using command `issubset`
- Test for superset using command `issuperset`
- Every set is a subset of itself
- The empty set is a subset of every set

```
1 A = {2, 4}
2 B = {1, 2, 3, 4, 5}
3 Is A a subset of B? True
4 Is B a subset of A? False
5 Is A a superset of B? False
6 Is B a superset of A? True
7 True
8 True
```

Outline

1. Defining Sets	2
1.1. Number Sets	5
1.2. Set Builder Notation	6
1.3. Venn Diagrams	10
1.4. Python Implementation	11
2. Relationships Between Sets	12
2.1. Subsets and Proper Subsets	14
2.2. Power Set	17
2.3. Python Implementation	19
3. Set Operations	20
3.1. Intersection, Union, Set Difference, ...	21
3.2. Cartesian product	27
3.3. Python Implementation	29
4. Proving Equivalence	31

Intersection

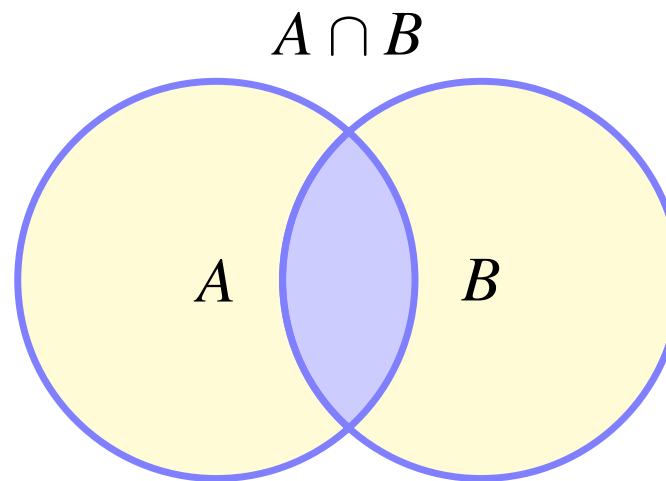
Definition 8 (Intersection)

The **intersection** of two sets, A and B , denoted by $A \cap B$, is the set that contains all elements that are elements of both A and B . We write

$$A \cap B = \{x \mid (x \in A) \wedge (x \in B)\}$$

Properties:

- $A \cap B = B \cap A$
- $(A \cap B) \cap C = A \cap (B \cap C)$
- $A \cap B = A \Rightarrow A \subseteq B$
- $A \cap \emptyset = \emptyset$
- Acts like the logical AND



Two sets are said to be **disjoint** if their intersection is empty

$$A \cap B = \emptyset$$

Union

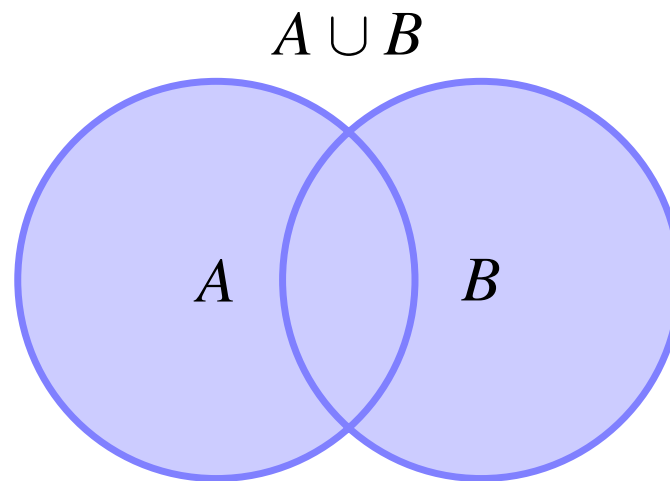
Definition 9 (Union)

The **union** of two sets, A and B , denoted by $A \cup B$, is the set that contains all elements that are elements of A or B or both. We write

$$A \cup B = \{x \mid (x \in A) \vee (x \in B)\}$$

Properties:

- $A \cup B = B \cup A$
- $(A \cup B) \cup C = A \cup (B \cup C)$
- $A \cup B = A \Rightarrow B \subseteq A$
- $A \cup \emptyset = A$
- Acts like the logical OR
- $|A \cup B| = |A| + |B| - |A \cap B|$



Set Difference

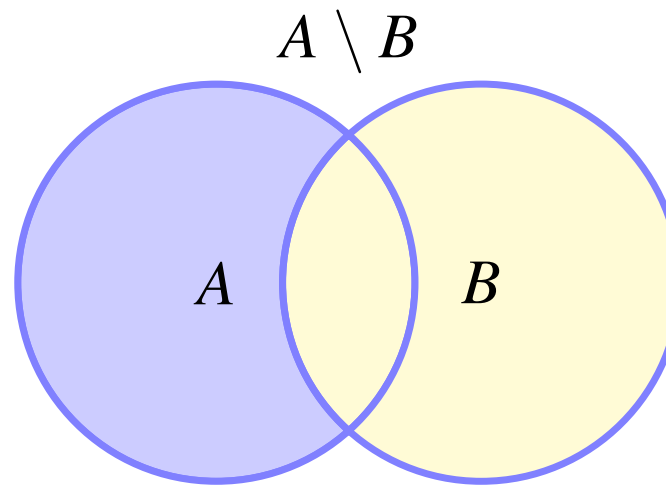
Definition 10 (Set Difference)

The **set difference** of two sets, A and B , denoted by $A \setminus B$, is the set that contains all elements that are in A but not in B . We write

$$A \setminus B = \{x \mid (x \in A) \wedge (x \notin B)\}$$

Properties:

- $A \setminus B \neq B \setminus A$
- $(A \setminus B) \setminus C \neq A \setminus (B \setminus C)$
- $A \setminus B = A \Rightarrow B \cap A = \emptyset$
- $A \setminus B = \emptyset \not\Rightarrow A = B$
- $A \setminus B = \emptyset \Rightarrow A \subseteq B$



Symmetric Difference

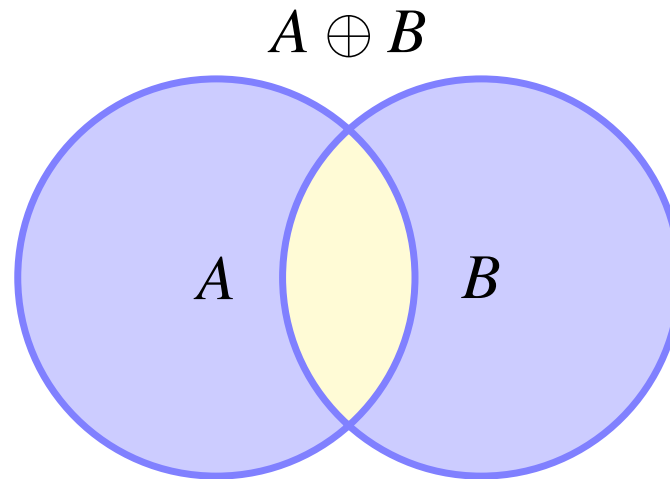
Definition 11 (Symmetric Difference)

The **symmetric difference** of two sets, A and B , denoted by $A \oplus B$, is the set that contains all elements that are in A or in B but not both. We write

$$A \oplus B = \{x \mid (x \in A \cup B) \wedge (x \notin A \cap B)\}$$

Properties:

- $A \oplus B = B \oplus A$
- $(A \oplus B) \oplus C = A \oplus (B \oplus C)$
- $A \oplus B = A \Rightarrow B = \emptyset$
- $A \oplus B = \emptyset \Rightarrow A = B$



Set Complement

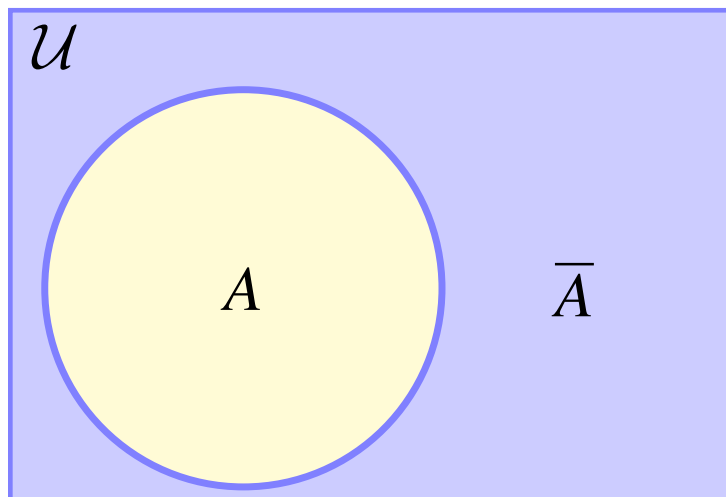
Often when dealing with sets, we will have some understanding as to what “everything” is. Perhaps we are only concerned with natural numbers. In this case we would say that our universe is \mathbb{N} . We denote this universe by \mathcal{U} . Given this context, we might wish to speak of all the elements which are not in a particular set.

Definition 12 (Complement)

The **complement** of a set A , denoted by \bar{A} , is the set containing all elements not in A .

Properties:

- $\bar{A} \cap A = \emptyset$
- $\bar{A} \cup A = \mathcal{U}$
- $A \cap \bar{B} = A \setminus B$



Example

Let $A = \{1, 2, 3, 4, 5, 6\}$, $B = \{2, 4, 6\}$, $C = \{1, 2, 3\}$ and $D = \{7, 8, 9\}$. If the universe is $\mathcal{U} = \{1, 2, \dots, 10\}$, find:

- a) $A \cup B$.
- b) $A \cap B$.
- c) $B \cap C$.
- d) $A \cap D$.
- e) $\overline{B \cup C}$.
- f) $A \setminus B$.
- g) $(D \cap \overline{C}) \cup \overline{A \cap B}$.
- h) $\emptyset \cup C$.
- i) $\emptyset \cap C$.

Cartesian Product

Given two sets we often need to construct a set of all possible pairing of elements from both sets.

Definition 13 (Cartesian Product)

The **Cartesian product** of two sets A and B , denoted by $A \times B$ is the set of **ordered pairs** where the first member is an element of the first set and the second member is an element of the second.

$$A \times B = \{(a, b) \mid a \in A, b \in B\}$$

- When we take the Cartesian product of a set, say A , by itself, we write A^2 .
- The 2D plane is the Cartesian product of the set of real numbers (\mathbb{R}) with itself.

Example

Example 14

Let $A = \{1, 2\}$ and $B = \{3, 4, 5\}$.

- a) Find $A \times B$ and $A \times A$.
- b) How many elements do you expect to be in $B \times B$?
- c) Is $A \times B = B \times A$?

Solution.

- a) $A \times B = \{(1, 3), (1, 4), (1, 5), (2, 3), (2, 4), (2, 5)\}$.
 $A \times A = A^2 = \{(1, 1), (1, 2), (2, 1), (2, 2)\}$.
- b) $|B \times B| = 9$. There will be 3 pairs with first coordinate 3, three more with first coordinate 4, and a final three with first coordinate 5.
- c) No. Cartesian product generates ordered pairs.

Python Implementation

set_operations.py

```
1 A = {1,2,3,4,5,6}
2 B = {2,4,6}
3 C = {1,2,3}
4 D = {7,8,9}
5 U = set(range(1,11))
6
7 ans = A.union(B)
8 print("A u B = ", ans)
9
10 ans = A.intersection(B)
11 print("A n B = ", ans)
12
13 ans = A.difference(B)
14 print("A \ B = ", ans)
15
16 ans = U.difference(A)
17 print("A complement", ans)
18
19 N = set()
20 print(N.union(C)==C)
21 print(N.intersection(C)==N)
```

- Create A , B , C , D and universal set U
- Use **union** (or $|$ operator) for set union
- Use **intersection** (or $\&$ operator) for set intersection
- Use **difference** (or $-$ operator) for set difference. Also have **symmetric_difference**
- Complement is implemented using set difference.
- Properties of operations hold as expected.

```
1 A u B = {1, 2, 3, 4, 5, 6}
2 A n B = {2, 4, 6}
3 A \ B = {1, 3, 5}
4 A complement {8, 9, 10, 7}
5 True
6 True
```

Outline

1. Defining Sets	2
1.1. Number Sets	5
1.2. Set Builder Notation	6
1.3. Venn Diagrams	10
1.4. Python Implementation	11
2. Relationships Between Sets	12
2.1. Subsets and Proper Subsets	14
2.2. Power Set	17
2.3. Python Implementation	19
3. Set Operations	20
3.1. Intersection, Union, Set Difference, ...	21
3.2. Cartesian product	27
3.3. Python Implementation	29
4. Proving Equivalence	31

Proving Equivalence

- We have mentioned a number of properties of set operations. Hopefully, some are obvious. How can we prove the others?
- We want some technique/process that will allow us to
 - Determine whether one set is a subset, a proper subset of another set.
 - Determine whether two sets are equal.
- We will use two[†] approaches:
 - on predicate logic based on the subset relationships

$$A \subseteq B \iff \forall x [x \in A \rightarrow x \in B]$$

$$A \subset B \iff \forall x [x \in A \rightarrow x \in B] \wedge \exists x [x \in B, x \notin A]$$

and

$$A = B \iff (A \subseteq B) \wedge (B \subseteq A)$$

- constructing a membership table.

[†]Well three approaches if we count using IPython to do the donkey work for us.

Example

Example 15

Let $A = \{x \mid x \text{ is even}\}$ and $B = \{x \mid x \text{ is a multiple of 3}\}$ and $C = \{x \mid x \text{ is a multiple of 6}\}$. Prove

$$A \cap B = C$$

Proof.

Proving $A \cap B \subseteq C$...

Let $x \in A \cap B$. Then x is a multiple of 2 and x is a multiple of 3. Therefore we can write

$$x = (2)(3)k \quad \text{for some integer } k$$

Hence x is a multiple of 6 and therefore $x \in C$.

Proving $C \subseteq A \cap B$...

Let $x \in C$. Then x is a multiple of 6 and so $x = 6k$ for some integer k , i.e.,

$$x = 6k = (2)(3)k$$

Therefore x is a multiple of 2 and a multiple of 3, and so $x \in A \cap B$. □

Example

Example 16 (Using Membership Tables)

Prove

$$\overline{A \cap B \cap C} = \bar{A} \cup \bar{B} \cup \bar{C}$$

Proof.

A membership table lists all possibilities of whether an element is in some sets or not ...

A	B	C	$A \cap B \cap C$	$\overline{A \cap B \cap C}$	\bar{A}	\bar{B}	\bar{C}	$\bar{A} \cup \bar{B} \cup \bar{C}$
F	F	F	F	T	T	T	T	T
F	F	T	F	T	T	T	F	T
F	T	F	F	T	T	F	T	T
F	T	T	F	T	T	F	F	T
T	F	F	F	T	F	T	T	T
T	F	T	F	T	F	T	F	T
T	T	F	F	T	F	F	T	T
T	T	T	T	F	F	F	F	F

Columns are identical, so given identity is **True**.

