Discrete Mathematics Course Notes

Felipe Balbi

October 22, 2019

Week 1

Learning objectives:

- Define a set, the elements of a set and the cardinality of a set.
- Define the concepts of the universal set and the complement of a set, and the difference between a set and a powerset of a set.
- Define the concepts of the union, intersection, set difference and symmetric difference, and the concept of a membership table.

1.101 Introduction to discrete mathematics

The study of discrete objects. Such objects are separated or distant from each other.

We will study integers, propositions, sets, relations or functions.

We will learn their properties and relationships among them.

Sets, functions, logic, graphs, trees, relations, combinatorics, mathematical induction and recursive relations. We will gain mathematical understanding of these topics and that will improve our skill of thinking in abstract terms.

1.104 The definition of a set

Set Theory deals with properties of well-defined collection of objects. Introduced by George Cantor.

Forms the basis of other fields of study: counting theory, relations, graph theory and finite state machines.

Definition of a set

A collection of any kind of objects: people, ideas, numbers. . .

A set must be well-defined, meaning that there can be no ambiguity to which objects belongs to the set.

$$\begin{aligned} E &= \{2,4,6,8\} \\ V &= \{a,e,i,o,u\} \\ EmptySet &= \{\} = \emptyset \end{aligned} \tag{1}$$

Definition 1 (Set)**.** *A set is an* ***unordered*** *collection of* ***unique*** *objects.*

Element of a set ($\in$)

Given the set $E = \{2,4,6,8\}$ we can say $2 \in E$ (2 is an element of E) and $3 \notin E$ (3 is not an element of E)

Cardinality of a set (Card)

Definition 2 (Cardinality)**.** *Given a set S, the* ***cardinality*** *of S is the number of elements contained in S. We write the cardinality of S as $|S|$. Note that the cardinality of the empty set is zero ($|\emptyset| = 0$)*

Subset of a set ($\subseteq$)

Definition 3 (Subset)**.** *A is said to be a subset of B if and only if every element of A is also an element of B. In this case we write $A \subseteq B$.*

This means we have the following equivalence:

$$A \subseteq B \iff \text{if x} \in A \,\text{then}\, x \in B (\text{for all x}) \tag{2}$$

The emptyset $\emptyset$ is a subset of any set.

Any set if a subset of itself ($S \subseteq S$)

Special Sets: $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{R}$

$\mathbb{N}$: set of natural numbers

$\mathbb{Z}$: set of integers

$\mathbb{Q}$: set of rational numbers

$\mathbb{R}$: set of real numbers

1.106 The listing method and rule of inclusion

Two different ways of representing a set.

The listing method consists of simply listing all elements of a set.

$S_1 = \{1,2,3\}$

The rule of inclusion method consists of producing a rule such that when that rule is true, the element is a member of the set. For example, here's a rule of inclusion for the set of all **odd** integers:

$S_2 = \{2n + 1 \mid n \in \mathbb{Z}\}$

In some cases, the rule of inclusion (or set building notation) is the only way to actually describe a set. For example, if we were to try to list the elements of the set of rational numbers $\mathbb{Q}$, we would never be able to reach the end. However, with the set builder notation it becomes simple and concise:

$\mathbb{Q} = \{\frac{n}{m} \mid n, m \in \mathbb{Z} \text{and} m \neq 0\}$

We can use the same notation for the set of elements in my bag:

$S_{bag} = \{x \mid x \text{is in my bag}\}$

1.108 The powerset of a set

A set can contain other sets as elements. For example:

$$\begin{aligned} A &= \{1, 2, 3, 4, 5, 6, 7, 8, 9\} \\ B &= \{\{1, 2, 3, 4\}, \{5, 6\}, \{7, 8, 9\}\} \end{aligned} \tag{3}$$

Note that $\{1, 2, 3, 4\}$ is a **subset** of A but it is an **element** of B. In mathematical terms:

$$\{1, 2, 3, 4\} \subseteq A \,\text{but}\, \{1, 2, 3, 4\} \in B \tag{4}$$

Powerset of a set

Definition 4 (Powerset)**.** *Given a set S, the powerset of S, $P(S)$, is the set containing **all** the **subsets** of S*

Example 1 Given a set $S = \{1, 2, 3\}$, the subsets of S are:

$\emptyset, \{1\}, \{2\}, \{3\},$
$\{1, 2\}, \{1, 3\}, \{2, 3\},$
$\{1, 2, 3\}$

Therefore, the powerset of S, $P(S)$ is as follows:

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

Example 2 What is the powerset of the empty set? What is the powerset of the powerset of the empty set?

$$\begin{aligned} P(\emptyset) &= \{\emptyset\} \\ P(P(\emptyset)) &= \{\emptyset, \{\emptyset\}\} \end{aligned} \tag{5}$$

Cardinality of a powerset

Given a set S, then $|P(S)| = 2^{|S|}$

In other words: the cardinality of the powerset of S is the 2 to the power of the cardinality of S. For example:

$$\begin{aligned} S &= \{1, 2\} \\ |S| &= 2 \\ P(S) &= \{\emptyset, \{1\}, \{2\}, \{1, 2\}\} \\ |P(S)| &= 4 = 2^2 = 2^{|S|} \end{aligned} \tag{6}$$

Example Given a set A, if $|A| = n$ find $|P(P(P(A)))|$

$$\begin{aligned} |P(A)| &= 2^n \\ |P(P(A))| &= 2^{2^n} \\ |P(P(P(A)))| &= 2^{2^{2^n}} \end{aligned} \tag{7}$$

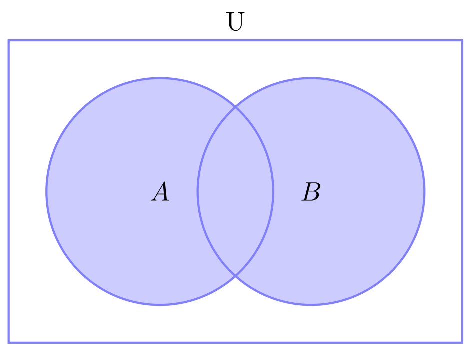
1.110 Set operations

We will look at set operations (intersection, union, difference, symmetric difference).

Union ($\cup$)

Definition 5 (Union)**.** *Given two sets A and B, the union of A and B, $A \cup B$, contains all the elements in **either** A or B.*

$$A \cup B = \{x \mid x \in A \,\mathrm{or}\, x \in B\} \tag{8}$$



Example

$$
\begin{aligned}
A &= \{1, 2, 3\} \\
B &= \{4, 5, 6\} \\
A \cup B &= \{1, 2, 3, 4, 5, 6\}
\end{aligned}
\tag{9}
$$

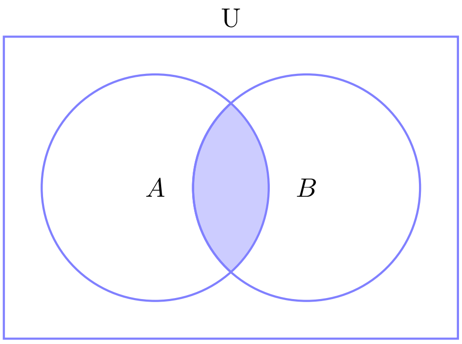
Membership Table $(A \cup B)$

A	B	$A \cup B$
0	0	0
0	1	1
1	0	1
1	1	1

Intersection $(\cap)$

Definition 6 (Intersection)**.** *Given two sets A and B, the intersection of A and B, $A \cap B$, contains all the elements in* ***both*** *A and B.*

$$A \cap B = \{x \mid x \in A \,\text{and}\, x \in B\} \tag{10}$$



;

Example

$$
\begin{aligned}
A &= \{1, 2, 3\} \\
B &= \{2, 3, 4\} \\
A \cap B &= \{2, 3, \}
\end{aligned}
\tag{11}
$$

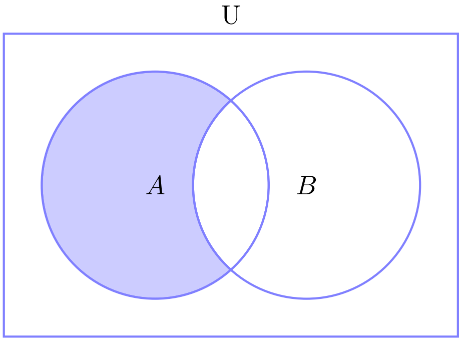
Membership Table $(A \cap B)$

A	B	$A \cap B$
0	0	0
0	1	0
1	0	0
1	1	1

Difference $(-)$

Definition 7 (Difference)**.** *Given two sets A and B, the difference of A and B, $A - B$, contains all the elements that are in A **but not** in B.*

$$A - B = \{x \mid x \in A \,\text{and}\, x \notin B\} \tag{12}$$



Example

$$\begin{aligned} A &= \{1, 2, 3\} \\ B &= \{3, 4, 5\} \\ A - B &= \{1, 2, \} \end{aligned} \tag{13}$$

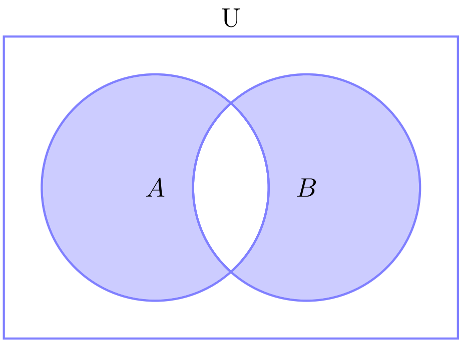
Membership Table $(A - B)$

A	B	$A - B$
0	0	0
0	1	0
1	0	1
1	1	0

Symmetric Difference $(\oplus)$

Definition 8 (Symmetric Difference)**.** *Given two sets A and B, the symmetric difference of A and B, $A \oplus B$, contains all the elements that are in A or in B **but not** in both.*

$$A \oplus B = \{x \mid (x \in A \,\text{or}\, x \in B) \,\text{and}\, x \notin A \cap B\} \tag{14}$$



Example

$$
\begin{aligned}
A &= \{1, 2, 3\} \\
B &= \{3, 4, 5\} \\
A \oplus B &= \{1, 2, 4, 5\}
\end{aligned} \tag{15}
$$

Membership Table ($A \oplus B$)

A	B	$A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

Summary

Operations

$$
\begin{aligned}
A &= \{1, 2, 3\} \\
B &= \{3, 4, 5\} \\
A \cup B &= \{1, 2, 3, 4, 5\} \\
A \cap B &= \{3\} \\
A - B &= \{1, 2\} \\
A \oplus B &= \{1, 2, 4, 5\}
\end{aligned} \tag{16}
$$

Membership Table

A	B	$A \cup B$	$A \cap B$	$A - B$	$A \oplus B$
0	0	0	0	0	0
0	1	1	0	0	1
1	0	1	0	1	1
1	1	1	1	0	0

Week 2

Learning objectives:

- Understand the concept of Venn diagrams and how they are used to represent and compare different set expressions.
- Understand and prove De Morgan's law using membership tables.

1.201 The representation of a set using Venn diagrams

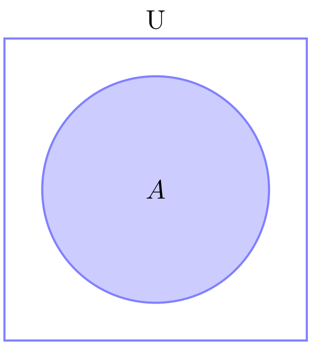
Venn diagrams can be used to represent sets and visualize the possible relations among a collection of sets. During this lesson we studied the following concepts:

- The universal set
- The complement of a set
- Set representation using Venn Diagrams

The Universal Set

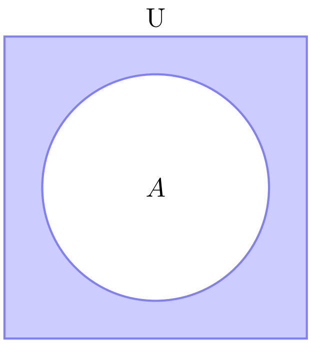
The universal set is a set containing everything. It's referred to by the letter `U`.

Note that $A \subseteq U$.



Complement of a set

Given a set A, the complement of A is written as $\overline{A}$, contains all the ements in the universal set U but not in A. It's represented by the area in red in figure below.



In other words $\overline{A} = U - A$.

Example

$$
\begin{aligned}
U &= \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\} \\
A &= \{2, 4, 6, 8, 10\} \\
\overline{A} &= U - A \\
&= \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\} - \{2, 4, 6, 8, 10\} \\
&= \{1, 3, 5, 7, 9\}
\end{aligned} \tag{17}
$$

The union of a set A with its completement $\overline{A}$ is always the universal set U.

$$A \cup \overline{A} = U \tag{18}$$

The symmetric difference of A and B is the same as the union of A and B minus the intersection of A and B:

$$A \oplus B = A \cup B - (A \cap B) \tag{19}$$

1.203 De Morgan's laws

De Morgan's laws describe how mathematical statements and concepts are related through their opposites. In se theory, they relate to intersection and unions of sets through their complements.

De Morgan's First Law

The complement of the union of two sets A and B is equal to the intersection of their complements.

$$\overline{A \cup B} = \overline{A} \cap \overline{B} \tag{20}$$

De Morgan's Second Law

The complement of the intersection of two sets A and B is equal to the union of their complements.

$$\overline{A \cap B} = \overline{A} \cup \overline{B} \tag{21}$$

Proof using membership tables

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

A	B	$\overline{A}$	$\overline{B}$	$A \cup B$	$\overline{A \cup B}$	$\overline{A} \cap \overline{B}$
0	0	1	1	0	1	1
0	1	1	0	1	0	0
1	0	0	1	1	0	0
1	1	0	0	1	0	0

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

A	B	$\overline{A}$	$\overline{B}$	$A \cap B$	$\overline{A \cap B}$	$\overline{A} \cup \overline{B}$
0	0	1	1	0	1	1
0	1	1	0	0	1	1
1	0	0	1	0	1	1
1	1	0	0	1	0	0

1.205 Laws of sets: Commutative, associative and distributive

We discussed three set identities: *Commutativity*, *Associativity*, and *Distributivity*.

Commutativity

When the order of operands in an operation does **NOT** affect the result, we say the operation is *commutative*. For example, addition is commutative

$$2 + 3 = 3 + 2 \tag{22}$$

Same applies for multiplication:

$$2 \cdot 3 = 3 \cdot 2 \tag{23}$$

Subtraction, however, is **NOT** commutative:

$$2 - 3 \neq 3 - 2 \tag{24}$$

In Set Theory, *Union* $\cup$, *Intersection* $\cap$, and *Symmetric Difference* $\oplus$ are all commutative operations. Much like in Algebra, Set difference is **NOT** commutative:

$$\begin{aligned} A &= \{1, 2\} \\ B &= \{1, 3\} \\ A - B &= \{1, 2\} - \{1, 3\} = \{2\} \\ B - A &= \{1, 3\} - \{1, 2\} = \{3\} \\ (A - B) &\neq (B - A) \end{aligned} \tag{25}$$

Associativity

When the grouping of elements in an operation doesn't change the result, we say the result is associative. Addition is associative:

$$(a + b) + c = a + (b + c) \tag{26}$$

In set theory, *Union*, *Intersection* and *Symmetric Difference* are all associative operations. Set difference is **not** associative:

$$\begin{aligned} A &= \{1, 2\} \\ B &= \{1, 3\} \\ C &= \{2, 3\} \\ (A - B) - C &= (\{1, 2\} - \{1, 3\}) - \{2, 3\} \\ &= \{2\} - \{2, 3\} \\ &= \emptyset \\ \\ A - (B - C) &= \{1, 2\} - (\{1, 3\} - \{2, 3\}) \\ &= \{1, 2\} - \{1\} \\ &= \{2\} \\ \\ &\therefore (A - B) - C \neq A - (B - C) \end{aligned} \tag{27}$$

Distributivity

The distributive property, in general, refers to the distributive law of multiplication which states that multiplying a sum of two numbers b and c by a coefficient a is the same as multiplying each addend by the coeficient a and adding the resulting products. We say the multiplication is distributive over the addition:

$$a \cdot (b + c) = a \cdot b + a \cdot c \tag{28}$$

Similarly, the set union is distributive over set intersection:

$$A \cup (B \cap C) = (A \cup B) \cap (B \cup C) \tag{29}$$

And the set intersection is distributive over the set union:

$$A \cap (B \cup C) = (A \cap B) \cup (B \cap C) \tag{30}$$

Table of Set Identities

Union	Name	Intersection
$A \cup B = B \cup A$	commutative	$A \cap B = B \cap A$
$(A \cup B) \cup C = A \cup (B \cup C)$	associative	$(A \cap B) \cap C = A \cap (B \cap C)$
$A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$	distributive	$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$
$\overline{A \cup B} = \overline{A} \cap \overline{B}$	De Morgan's Laws	$\overline{A \cap B} = \overline{A} \cup \overline{B}$
$A \cup \emptyset = A$	identities	$A \cap \emptyset = \emptyset$
$A \cup U = U$		$A \cap U = A$
$A \cup \overline{A} = U$	complement	$A \cap \overline{A} = \emptyset$
$\overline{U} = \emptyset$		$\overline{\emptyset} = U$
$\overline{\overline{A}} = A$	double complement	
$A \cup (A \cap B) = A$	absorption	$A \cap (A \cup B) = A$
$A - B = A \cap \overline{B}$	set difference	

Applying set identities to simplify expressions

Show that $\overline{(A \cap B) \cup \overline{B}} = B \cap \overline{A}$

$$\begin{aligned}
\overline{(A \cap B) \cup \overline{B}} &= \overline{(A \cap B)} \cap \overline{\overline{B}} \\
&= \overline{(A \cap B)} \cap B \\
&= (\overline{A} \cup \overline{B}) \cap B \\
&= \overline{A} \cap B \cup \overline{B} \cap B \\
&= \overline{A} \cap B \cup \emptyset \\
&= \overline{A} \cap B \\
&= B \cap \overline{A}
\end{aligned} \tag{31}$$

1.207 Partition

A partition of an object is a subdivision of the object into parts such that the parts are completely separated from each other, yet together they form the whole object.

Data partitioning has many applications in Computer Science such as Big Data analysis. This is usually referred to as *Divide and Conquer* approach. Such techniques must be applied in cases where the entire input data doesn't fit into the physical memory of the Computer. In such cases, we must find a way to partition the data so that subsets of the original data can be operated on without changing the result of the whole computation.

Definition of a partition of a set

Two sets A and B are said to be disjointed if and only if $A \cap B = \emptyset$.
Definition 9 (Set Partition)**.** *A partition of set A is a set of subsets A_i such that all subsets are disjointed and then union of all subsets A_i is equal to A.*

Week 3

Learning objectives:

- Define a function.
- Describe the properties of functions.
- Explain how to plot a function.

2.101 Introduction

A function is a rule that relates to how one quantity depends on another quantity. Much like a voltage depends on electrical current and resistance.

During this lecture, we learn the definition of a function and study a few of their properties.

2.102 The Definition of A Function

A function is a relation between a set of inputs and a set of outputs such that each input maps to exactly **one** output.

Definition

A function maps an element of set 1 to an element in set 2. Such mapping is *well-behaved* meaning that given a starting point we always know exactly where to go. For example, we could have a function that maps a set of strings to their corresponding number of characters:

$$\begin{aligned} S_1 &= \{Sea, Land, Sky\} \\ S_2 &= \{1, 2, 3, 4, 5, 6\} \end{aligned}$$

$$\begin{aligned} Sea &\rightarrow 3 \\ Land &\rightarrow 4 \\ Sky &\rightarrow 3 \end{aligned} \tag{32}$$

From Rosen's book, functions are defined as:
Definition 10 (Function)**.** *Let A and B be nonempty sets. A function f from A to B is an assignment of exactly one element of B to each element of A. We write $f(a) = b$ if b is the unique element of B assigned by the function f to the element a of A. If f is a function from A to B, we write $f : A \to B$ and read as f maps A to B.*

$x \in A : x \to f(x) = y \, (y \in B)$

Domain, co-domain and range of a function

Given a function $f : A \to B$

$x \in A \to f(x) = y \in B$

A is the set of inputs and its referred to as the *Domain of f*. We write it as $D_f = A$.

B is the set containing all possible outputs; referred to as the *co-domain of f*. We write it as $co - D_f = B$.

The set containing all outputs is called the *Range of f* and is written as R_f.

Image and pre-image (antecedent) of an element

y, the output of the function of a given input x, is called the *Image of* x where x itself is called the *pre-image of y*. We write $f(x) = y$.

Example of Domain, co-domain and range

Let A be the set $\{On, Sea, Land, Sky\}$, B be the set $\{1, 2, 3, 4, 5, 6\}$, and f be the function that maps the set of strings to their corresponding number of characters. We have:

$$\begin{aligned} On &\to 2 \\ Sea &\to 3 \\ Land &\to 4 \\ Sky &\to 3 \end{aligned} \tag{33}$$

In this case:

$$\begin{aligned} D_f &= A = \{On, Sea, Land, Sky\} \\ co - D_f &= B = \{1, 2, 3, 4, 5, 6\} \\ R_f &= \{2, 3, 4\} \end{aligned} \tag{34}$$

Moreover, we can say that 2 is the image of the string On and On is the pre-image of 2. $Pre - images(2) = \{On\}$.

3 is the image of *Sea* and *Sky*, therefore $Pre - images(3) = \{Sea, Sky\}$.

2.104 Plotting functions

We explore and plot some special functions.

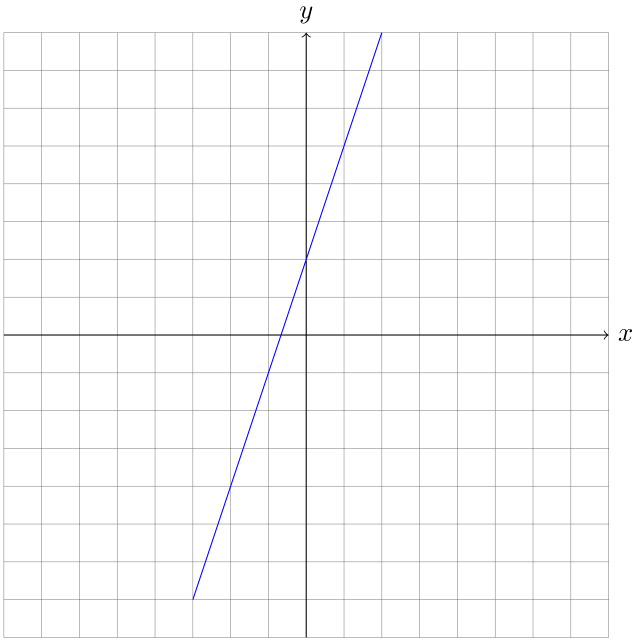
Linear Functions

A function f is called a linear function if it is of the form $f(x) = ax + b$. This function is a straight line passing through the point $(0, b)$ with gradient a.

If $a > 0$, then the function is increasing. It's decreasing if $a < 0$.

In order to plot this function, first we make a table of values for this function. We use $f(x) = 3x + 2$ as an example.

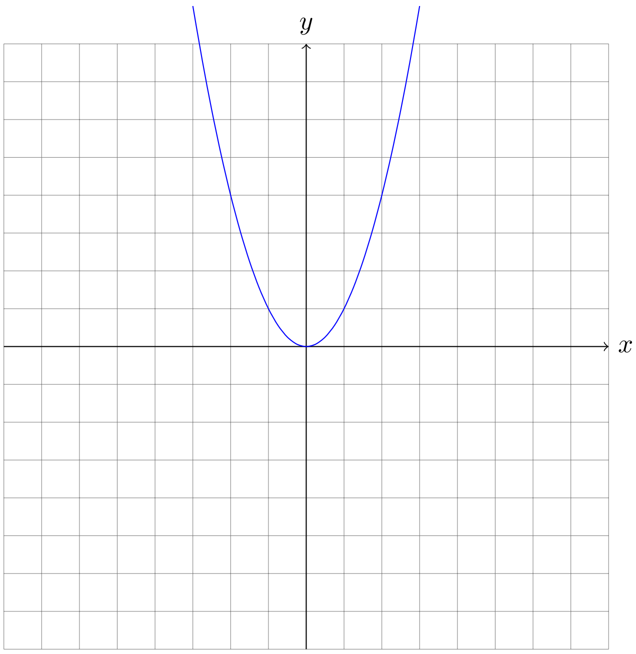
x	f(x)
0	2
1	5
2	8
3	11
4	14



Quadratic functions

A function f of the form $f(x) = ax^2 + bx + c$ is called a *Quadratic function.*

x	f(x)
0	0
1	1
2	4
3	9
4	16

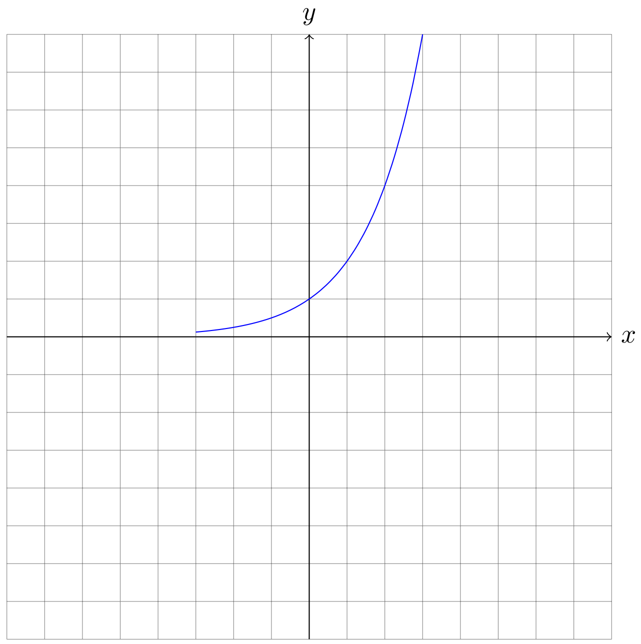


Exponential functions

A function f of the form $f(X) = b^x$ is called an *exponential function*. The variable b is called the *base* of the function.

A more formal definition may be:
Definition 11 (Exponential Function)**.** *The function f defined by $f : \mathbb{R} \to \mathbb{R}^+$ and $f(x) = b^x$ where $b > 0$ and $b \neq 1$ is called and exponential function with a base b.*

x	f(x)
0	1
1	2
2	4
3	8
4	16



Exponentials have some properties which are good to remember:

Form	Result
$b^x \cdot b^y$	b^{x+y}
$\frac{b^x}{b^y}$	b^{x-y}
$(b^x)^y$	$b^{x \cdot y}$
$(a \cdot b)^x$	$x^x \cdot b^x$
$\left(\frac{a}{b}\right)^x$	$\frac{a^x}{b^x}$
b^{-x}	$\frac{1}{b^x}$

The point $(0, 1)$ is the common point for all exponentials. When $b > 1$ we have an exponential growth. When $0 < b < 1$, we have exponential decay.

2.106 Injective and surjective functions

Injective Functions

Let $f : A \to B$ be a function; f is said to be injective, or *one-to-one* if and only if $\forall a, b \in A$, if $a \neq b$ then $f(a) \neq f(b)$. In plain english, this means that two different inputs will lead to two different outputs, i.e. given two different inputs a and b, then the **image** of a is different than the image of b.

A corollary of this is that:
Corollary 1. $\forall a, b \in A, f(a) = f(b) \implies a = b$

Example: linear function Show that a function $f : \mathbb{R} \to \mathbb{R}$ with $f(x) = 2x + 3$ is an injection (one-to-one).

We can prove this in two different ways. The first proof assumes \$f(a) $= f(b)$

Proof. Let $a, b \in \mathbb{R}$, show that if $f(a) = f(b)$ then $a = b$.

$$\begin{aligned} f(a) = f(b) \implies & 2a + 3 = 2b + 3 \\ & 2a + 3 - 3 = 2b + 3 - 3 \\ & 2a = 2b \\ & \frac{2a}{2} = \frac{2b}{2} \\ & a = b \end{aligned} \tag{35}$$

$\therefore f$ is injective. □

The second proof assumes $a \neq b$

Proof. Let $a, b \in \mathbb{R}$, show that if $a \neq b$ then $f(a) \neq f(b)$.

$$\begin{aligned} a \neq b \implies & 2a \neq 2b \\ & 2a + 3 \neq 2b + 3 \\ & f(a) \neq f(b) \end{aligned} \tag{36}$$

$\therefore f$ is injective. □

Example: quadratic function To prove that a function is not injective, we only need to find one example of two different inputs having the same image.

Show that a function $f : \mathbb{R} \to \mathbb{R}$ with $f(x) = x^2$ is not injective.

Proof.

$$\begin{aligned} f(5) = (5)^2 = (-5)^2 = f(-5) \\ \text{however}\, 5 \neq -5 \end{aligned} \tag{37}$$

$\therefore f$ is not injective. □

However, if we change the domain of the function such that $f : \mathbb{R}^+ \to \mathbb{R}$, we can make it injective. To prove this, we can apply the same two methodologies from the previous example.

Surjective Functions

Let $f : A \to B$ be a function; f is said to be surjective, or *onto* if and only if $\forall y \in B\, \exists x \in A \mid y = f(x)$. This means that every element in the co-domain of f, B, has **at least** one pre-image in the domain of f, A. This is equivalent to saying that the range and the co-domain of a surjective function, are equal (i.e. $R_f = co - D_f$).

Example: linear function Show that a function $f : \mathbb{R} \to \mathbb{R}$ with $f(x) = 2x + 3$ is a surjection (onto).

To prove this, we must show that for every element in B, there is a pre-image in A.

Proof. Let $y \in \mathbb{R}$, show that $\exists x \in \mathbb{R} \mid f(x) = y$.

$$\begin{aligned} f(x) = y \implies 2x + 3 = y \\ 2x + 3 - 3 = y - 3 \\ \frac{2x}{2} = \frac{y-3}{2} \\ x = \frac{y-3}{2} \in \mathbb{R} \end{aligned} \tag{38}$$

$\therefore \forall y \in \mathbb{R} \exists x = \frac{y-3}{2} \in \mathbb{R} \mid f(x) = y$, hence f is surjective. □

Example: quadratic function Show that a function $f : \mathbb{R} \to \mathbb{R}$ with $f(x) = x^2$ is not a surjection.

Proof. Let $y \in \mathbb{R}$, show that $\exists x \in \mathbb{R} \mid f(x) = y$.

Let $y \in \mathbb{R}$, show that $\exists x \in \mathbb{R} \mid f(x) = y$.

$R_f = [0, +\infty[\neq co - D_f = \mathbb{R}$

$\therefore f$ is not surjective. □

Week 4

Learning objectives:

- Discuss special functions.
- Describe inverse functions.

2.201 Function composition

Using examples we will understand function composition and how to work out the composition of two functions. We will also show that function composition is **not** commutative.

Given two functions, f and g, the composition of f and g is written as $f \circ g = f(g(x))$.

For example, let $f(x) = 2x$ and $g(x) = x^2$, the composition of f and g can be worked out as follows:

$$
\begin{aligned}
(f \circ g)(x) &= f(g(x)) \\
&= f(x^2) \\
&= 2x^2
\end{aligned}
$$

$$
\begin{aligned}
(f \circ g)(1) &= f(g(x)) \\
&= f(1^2) \\
&= 2 \cdot 1^2 \\
&= 2
\end{aligned}
$$

What this means is that if we have a function $g : A \to B$ and a function $f : B \to C$, function composition allows us to produce a function $(f \circ g) : A \to C$.

Note that function composition is **not** commutative. In other words, $f \circ g \neq g \circ f$. Let $f = 2x$ and $g = x^2$, we can show that $(f \circ g) = 2x^2$ and $(g \circ f) = 4x^2$.

2.203 Bijective functions

Definition

A bijective or invertible function is a function $f : A \to B$ that can be described as both *injective* and *surjective* simultaneously. This means that each element of the *co-domain* has exactly one *pre-image*.
Definition 12 (Bijection)**.** *A function $f(x)$ is said to be bijective if and only if it is both injective and surjective.*

Exercise 1:

Show that the function $f : \mathbb{R} \to \mathbb{R}$ with $f(x) = 2x + 3$ is a bijective (invertible) function.

Proof. To prove this, suffices to prove that this function is both an injection and a surjection. Let's prove the injection case first:

Let $a, b \in \mathbb{R}$, we will show that if $f(a) = f(b)$ then $a = b$.

$$
\begin{aligned}
f(a) = f(b) \implies & 2a + 3 = 2b + 3 \\
& 2a + 3 - 3 = 2b + 3 - 3 \\
& 2a = 2b \\
& \frac{2a}{2} = \frac{2b}{2} \\
& a = b
\end{aligned}
$$

$\therefore f$ is injective.

Now turning our attention to the surjection case, we have:

Let $y \in \mathbb{R}$, we will show that $\forall y \in \mathbb{R} \exists x \in \mathbb{R} \mid f(x) = y$.

$$\begin{aligned} f(x) = y \implies & 2x + 3 = y \\ & 2x + 3 - 3 = y - 3 \\ & \frac{2x}{2} = \frac{y-3}{2} \\ & x = \frac{y-3}{2} \in \mathbb{R} \end{aligned}$$

$\therefore \forall y \in \mathbb{R} \exists x = \frac{y-3}{2} \in \mathbb{R} \mid f(x) = y$, hence f is surjective.

Because we have proved that $f(x) = 2x+3$ is both an injection and a surjection, we have also proved that it is a bijection. □

Inverse function

Definition 13 (Inverse function)**.** *Let $f : A \to B$, if f is bijective, then the inverse function f^{-1} exists and is defined as $f^{-1} : B \to A$.*

Given this definition, let's find the inverse of $2x + 3$.

Exercise 2:

The following function $f : \mathbb{R} \to \mathbb{R}$ with $f(x) = 2x + 3$ is a bijection. Find the inverse function f^{-1}.

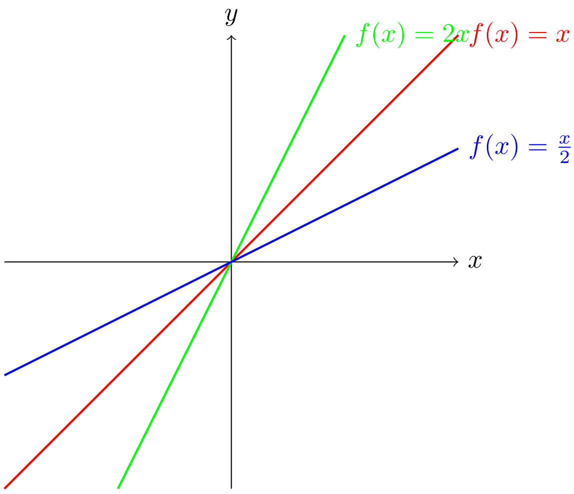
$$\begin{aligned} f(x) &= 2x + 3 \\ f(x) &= y \\ 2x + 3 &= y \\ 2x + 3 - 3 &= y - 3 \\ \frac{2x}{2} &= \frac{y-3}{2} \\ x &= \frac{y-3}{2} \\ \\ \therefore f^{-1}(x) &= \frac{x-3}{2} \end{aligned}$$

Identity function

There is one special case of composition which is $(f \circ f^{-1})(x) = (f^{-1} \circ f)(x) = x$. For example if $f(x) = 2x$, then $f^{-1}(x) = \frac{x}{2}$, therefore $(f \circ f^{-1})(x) = 2\frac{x}{2} = x$. Similarly, $(f^{-1} \circ f)(x) = \frac{2x}{2} = x$.

Plotting the inverse function

The function f and its inverse f^{-1} are always symmetric to the straight line $y = x$.



2.205 Logarithmic functions

Exponential and logarithmic functions are closely related. Therfore, let's review exponential functions before dealing with logarithmic functions.

Exponential functions were defined back in Definition 11. We know from that definition that:

$$y = f(x) = b^x \; (b > 0, \; b \neg 1)$$

The domain of the function is $(-\infty, +\infty)$.

The range of the function is $(0, +\infty)$.

The graph of an exponential function **always** passes through the point with coordinates $(0, 1)$. If the base b is greater than 1, then the function is increasing on $(-\infty, +\infty)$ and we call it *exponential growth*. Conversely, if $b < 1$, then the function is decreasing on $(-\infty, +\infty)$ and we call it *exponential decay*.

Definition

With that review out of the way, we can define Logarithmic functions:
Definition 14 (Logarithmic function)**.** *The logarithmic function with base b where $b > 0$ and $b \neq 1$ is defined as follows:*

$$log_b x = y \iff x = b^y$$

We can say that $log_b x$ is the inverse function of the exponential function b^x.

Laws of logarithmic functions

1. $log_b m \times n = log_b m + log_b n$
2. $log_b \frac{m}{n} = log_b m - log_b n$
3. $log_b m^n = n\ times log_b m$
4. $log_b 1 = 0$
5. $log_b b = 1$

Exercise 1

$log_3 81 \quad log_3 81 = log_3 3^4 = 4 \times log_3 3 = 4 \times 1 = 1$

$log_{10} 100 \quad log_{10} 100 = log_{10} 10^2 = 2 \times log_{10} 10 = 2 \times 1 = 2$

$log_3 \frac{1}{81} \quad log_3 \frac{1}{81} = log_3 81^{-1} = log_3 3^{-4} = -4 \times log_3 3 = -4 \times 1 = -4$

$log_2 1 \quad log_2 1 = log_2 2^0 = 0 \times log_2 2 = 0 \times 1 = 0$

Natural logarithm

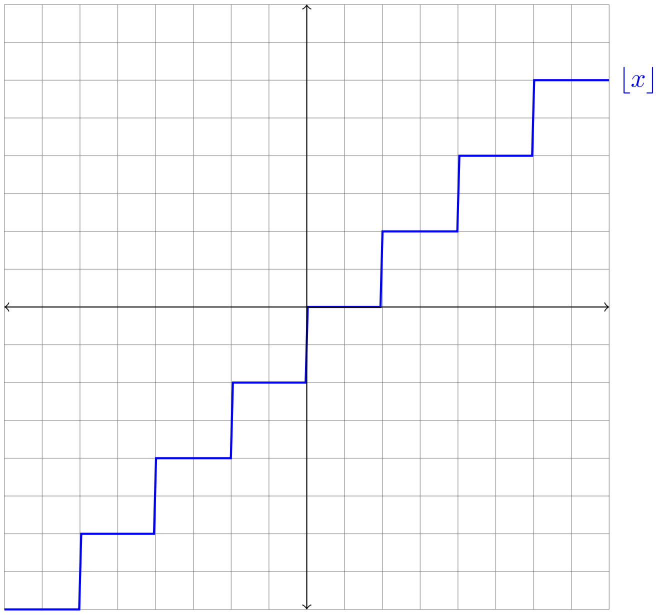
The natural logarithm, commonly written as $ln(x)$ is the logarithm with base e. In other words: $ln(x) = log_e x$ where $e \approx 2.71828$.

2.207 Floor and ceiling functions

Floor function

Definition 15 (Floor function)**.** *The* ***floor*** *function is a function* $f : \mathbb{R} \to \mathbb{Z}$. *It takes a real number* x *as input and outputs the largest integer that is less than or equal to* x. *Denoted as* $floor(x) = \lfloor x \rfloor$.

For example, given a real number x such that $n \leq x < n + 1$, the floor of x is n. In other words: $floor(x) = \lfloor x \rfloor = n$.

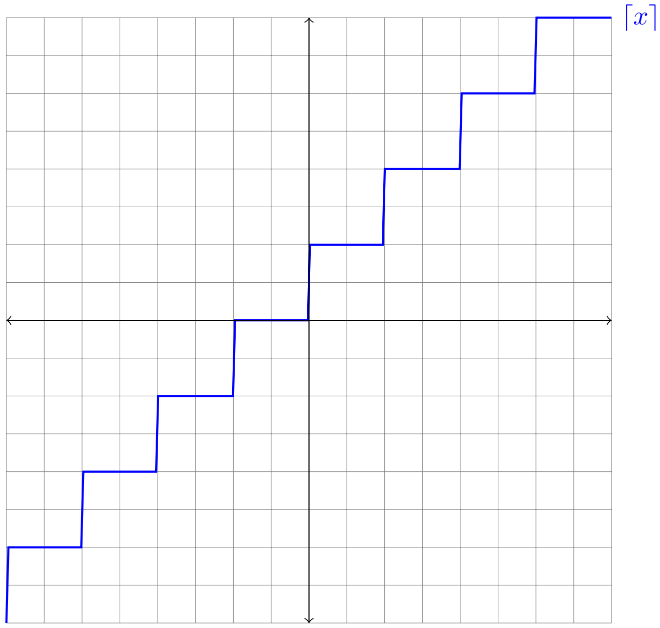


We can think of the floor function as if we're walking on the number line to the left until we find an integer. This means that $\lfloor 1.1 \rfloor = 1$ but $\lfloor -1.1 \rfloor = -2$.

Ceiling function

Definition 16 (Ceiling function)**.** *The **ceiling** function is a function* $f : \mathbb{R} \to \mathbb{Z}$*. It takes a real number* x *as input and outputs the smallest integer that is greater than or equal to* x*. Denoted as* $ceiling(x) = \lceil x \rceil$*.*

For example, given a real number x such that $n < x \leq n + 1$, the ceiling of x is $n + 1$. In other words: $ceiling(x) = \lceil x \rceil = n + 1$.



This is exact opposite of the floor function. So we can think of it as if were were walking on the number line to the right until we find an integer. This means that $\lfloor 1.1 \rfloor = 2$, but $\lfloor -1.1 \rfloor = -1$.

Exercise 1

Let n be an integer and x a real number. Show that:

$$\lfloor x + n \rfloor = \lfloor x \rfloor + n$$

Proof. Let m be an integer such that $m = \lfloor x \rfloor$. By definition of the floor function we have $m \leq x < m + 1$. Addin n to both sides of this inequality, we have $m + n \leq x + n < m + n + 1$.

This implies that $\lfloor x + n \rfloor = m + n$ by definition. And $m = \lfloor x \rfloor$. Therefore $\lfloor x + n \rfloor = \lfloor x \rfloor + n$. □

Week 5

Learning Objectives

- Explain and apply basic concepts of propositional logic.
- Construct truth tables of propositions and use them to demonstrate the equivalence of logical statements.
- Translate natural language statements into symbolic logical statements and vice versa.

3.101 Introduction to propositional logic

Definition 17 (Propositional Logic)**.** *It is a branch of logic that is interested in studying mathematical statements.*

Propositional Logic is the basis of all mathematical reasoning and the rules used to construct mathematical theories. Its original purpose was to model reasoning and dates back to Aristotle.

Effectively, it is an *algebra of propositions.* In this *algebra*, the variables are unknown **propositions** rather than unknown **real numbers**.

The operators used are and ($\wedge$), or ($\vee$), not ($\neg$), implies ($\implies$) and if and only if ($\iff$) instead of our regular $+$, $-$, $\times$, and $\div$.

Applications of propositional logic

Propositional logic can be used in logic circuit design. It can also be applied to programming languages, such as Prolog.

Many computer reasoning systems, including theorem provers, program verifiers and applications in the field of Artificial Intelligence, have been implemented in logic-based programming languages.

These languages, generally employ predicate logic, a form of logic that extends the capabilities of propositional logic.

3.103 Propositions

Definition 18 (Proposition)**.** *A declarative sentence that is either true or false, but not both.*

A Proposition is the most basic element of logic. Which means that propositions are the building blocks for our reasoning and logical statements.

Examples of propositions

As mentioned above, a proposition must be a declarative statement that is either true or false, therefore the following statements are propositions:

- **London is the capital of the United Kingdom**

  We know this is true, so this is considered to be a **true proposition**.

- $\mathbf{1 + 1 = 2}$

  This is also a **true proposition**.

- $\mathbf{2 < 3}$

  This is also a **true proposition**.

- **Madrid is the capital of France**

  This is a **false proposition**.

- $\mathbf{3 < 2}$

  This is also a **false proposition**.

- **10 is an odd number**

  This is also a **false proposition**.

What follows is a series of statements which are **not** propositions, as they can not assume a true or false value:

- $x + 1 = 2$

  We don't know the value of x, so this is **not** a proposition. However, if a value is assigned to x, then at that moment it becomes a proposition. IF we assign the value 1 to x, this will be a **true** proposition, if any other value is assigned to x, then it'll be a **false** proposition.

- $x + y = z$

  Also not a proposition as x, y and z have no values.

- What time is it?

  Is not a proposition as it is not a declarative sentence.

Propositional Variables

To avoid writing long, repetivive propositions, we make use of **propositional variables**. They are typically a letter, such as **p**, **q**, **r**, ...

We can assign letters to our previous propositions, for example: Let p be the proposition *London is the capital of the United Kingdom.*

3.105 Truth tables and truth sets

As we begin to build more complex compound propositions, we need a method of keeping track of this proposition's truth value. A truth table is one such method.

True Tables

A truth table is tabular representation of all the possible combinations of truth values for a set of propositional variables.

For example:

p	q
F	F
F	T
T	F
T	T

A truth table of n propositional variables, will contain 2^n rows. So a table for 3 propositional variables, will have 8 rows:

p	q	r
F	F	F
F	F	T
F	T	F
F	T	T
T	F	F
T	F	T
T	T	F
T	T	T

Truth Set

Definition 19 (Truth Set)**.** *Let p be a proposition on a set S. The truth set of p is the set of elements of S for which p is true.*

Commonly, we use a capital letter to refer to the truth set of a proposition. For example the truth set of a proposition p is referred to as P.

Example Let $S = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$.

Let p and q be two propositions concerning an integer n in S, defined as follows:

$p : n$ is even
$q : n$ is odd

Therefore, the truth set of p is $P = \{2, 4, 6, 8, 10\}$ and the truth set of q is $Q = \{1, 3, 5, 7, 9\}$.

3.107 Compound propositions

Compound propositions are built by combining propositions with logical operators (also referred to as connectives). The connectives which we deal with in this lecture are:

- **Negation** $\neg$
- **Conjunction** $\wedge$
- **Disjunction** $\vee$
- **Exclusive-or** $\oplus$

Negation $\neg$

Let p be a proposition, the negation of p, denoted by $\neg p$, and read as "not p", is the statement: *it is **not** the case that p.*

For example if p is the statement *John's program is written in Python*, then $\neg p$ is the statement *John's program is **not** written in Python.*

p	$\neg p$
F	T
T	F

Conjunction $\wedge$

Let p and q be propositions, the conjuntion of p and q, denoted by $p \wedge q$, and read as "p and q", is the statement: *p and q.*

The conjunction is only true when both p and q are true and false otherwise.

For example if p is the statement *John's program is written in Python*, and q is the statement *John's program has less then 20 lines of code*, then $p \wedge q$ is the statement *John's program is written in Python **and** has less than 20 lines of code.*

p	q	$p \wedge q$
F	F	F
F	T	F
T	F	F
T	T	T

Disjunction $\vee$

Let p and q be propositions, the disjuntion of p and q, denoted by $p \vee q$, and read as "p or q", is the statement: *p or q.*

The disjunction is only false when both p and q are false and true otherwise.

For example if p is the statement *John's program is written in Python*, and q is the statement *John's program has less then 20 lines of code*, then $p \vee q$ is the statement *John's program is written in Python **or** has less than 20 lines of code.*

p	q	$p \vee q$
F	F	F
F	T	T
T	F	T
T	T	T

Exclusive-or $\oplus$

Let p and q be propositions, the exclusive-or of p and q, denoted by $p \oplus q$, and read as "p exclusive-or q", is the statement: *p exclusive-or q.*

The exclusive is true when either p or q are true, but not both.

For example if p is the statement *John's program is written in Python*, and q is the statement *John's program has less then 20 lines of code*, then $p \oplus q$ is the statement *John's program is written in Python* ***or*** *has less than 20 lines of code,* ***but not both***.

p	q	$p \oplus q$
F	F	F
F	T	T
T	F	T
T	T	F

Precedence of logical operators

Propositions can be combined to build complex compound propositions. To do this we need to start relying on precedence of logical operators or use parenthesis.

The meaning of compound propositions can change depending on the order in which parentheses are used. For example $(p \lor q) \land (\neg r) \neq p \lor (q \land \neg r)$.

Here's a small table of precedence:

Operator	Precedence
$\neg$	1
$\land$	2
$\lor$	3

Exercise

Given a positive integer n, let's consider the propositions p and q, where:

- p: *n is an even number*
- q: *n is less than 10*

Let's write the logical expression for each of the following propositions:

1. *n is an even number* **and** *is less than 10* $p \land q$
2. *n is either an even number* **or** *is less than 10* $p \lor q$
3. *n is either an even number* **or** *is less than 10* **but not both** $p \oplus q$
4. $\neg p \lor (p \land q)$

p	q	$p \land q$	$p \lor q$	$p \oplus q$	$\neg p$	$\neg p \lor (p \land q)$
F	F	F	F	F	T	T
F	T	F	T	T	T	T
T	F	F	T	T	F	F
T	T	T	T	F	F	T

Week 6

Learning Objectives

- How to formalise a logical implication
- Apply the laws of propositional to analyse propositions and arguments.

3.202 Logical implication ($\implies$)

Definition 20 (Implication)**.** *Let p and q be propositions. The conditional statement or implication $p \implies q$ is the proposition "if p then q".*

p is called the hypothesis (or antecedent) q is called the conclusion (or consequence)

Example 1 Let p and q be the following statements:

- p: John did well in Discrete Mathematics
- q: John will do well in the Programming Course

The conditional statement $p \implies q$ can be written as follows:

If John did well in Discrete Mathematics then John will do well in the Programming Course.

Truth Table

p	q	$p \implies q$
F	F	T
F	T	T
T	F	F
T	T	T

As we can see, the only situation where an implication evaluates to false is when our hypothesis is true but the conclusion is false.

Different expressions for $p \implies q$

Let p and q be the following statements:

- p: It's sunny
- q: John goes to the park

There are many ways to write the conditional statement $p \implies q$:

- $p \implies q$
- if p **then** q
- if p, q

- p implies q
- p only if q
- q follows from p
- p is sufficient for q
- q unless $\neg p$
- q is necessary for p

All of these forms are equivalent to *if it's sunny then John goes to the park.*

Note that the statement *John going to the park is necessary for a sunny day* sounds a bit strange in English. We should try to think of it as *John going to the park is a necessary consequence of a sunny day.*

Converse, inverse and contrapositive

Let p and q be propositions and A the conditional statement $p \implies q$.

The conditional statement $q \implies p$ is referred to as the **converse** of A.

The conditional statement $\neg q \implies \neg p$ is referred to as the **contrapositive** of A.

The conditional statement $\neg p \implies negq$ is referred to as the **inverse** of A.

The **contrapositive** of A has the same truth table as A and is, therefore, equivalent to it.

Example 2 Let p and q be the following statements:

- p: It's sunny
- q: John goes to the park
- $A = p \implies q$: If it's sunny then John goes to the park

Therefore:

- **converse**: If John goes to the park, then it's sunny
- **contrapositive**: If John does not go to the park, then it's not sunny
- **inverse**: If it's not sunny then John does not go the park

We can build a large truth table with all of these:

p	q	$p \implies q$	$\neg q \implies \neg p$	$\neg p \implies \neg q$	$q \implies p$
F	F	T	T	T	T
F	T	T	T	F	F
T	F	F	F	T	T
T	T	T	T	T	T

Note, also, that the **converse** or A and the **inverse** of A are equivalent.

3.204 Logical equivalence ($\iff$)

Definition 21 (Logical Equivalence)**.** *Let p and q be propositions. The **biconditional** or **equivalence** statement $p \iff q$ is the proposition $p \implies q \wedge q \implies p$.*

Biconditional statements are also called bi-implications and can be read p ***if and only if*** q

Truth Table

\$p	q	$p \iff q$
F	F	T
F	T	F
T	F	F
T	T	T

We can see here that the biconditional statement of p and q is true whenever p and q have the same truth value and is false otherwise.

Equivalent propositions

Let p and q be propositions. We say that p and q are *logically equivalent* if they always have the same truth value.

We write $p \equiv q$ to signify that p is equivalent to q.

Note that $\equiv$ is not a logical operator, and $p \equiv q$ is not a compound proposition. $p \equiv q$ means that the compound proposition $p \iff q$ is always true.

Proving equivalence

One way of determining logical equivalence, is by means of truth tables and verifying that two propositions have the same truth values for every possible input.

p	q	$p \implies q$	$\neg p$	$\neg p \vee q$
F	F	T	T	T
F	T	T	T	T
T	F	F	F	F
T	T	T	F	T

If values difer in any row, then we demonstrate non-equivalence.

Example 1 Let p, q, and r be the following propositions concerning n:

- p: $n = 20$
- q: n is even
- r: n is positive

Let's express each conditional statement below symbolically:

- **If** $n = 20$, **then** n is positive

  $p \implies r$

- $n = 20$ **if** n is even

  $q \implies p$

- $n = 20$ **only if** n is even

  $p \implies q$

Precedence of logical operators (Updated)

Propositions can be combined to build complex compound propositions. To do this we need to start relying on precedence of logical operators or use parenthesis.

The meaning of compound propositions can change depending on the order in which parentheses are used. For example $(p \lor q) \land (\neg r) \neq p \lor (q \land \neg r)$.

Here's a small table of precedence:

Operator	Name	Precedence
$\neg$	Negation	1
$\land$	Conjunction	2
$\lor$	Disjunction	3
$\implies$	Conditional or Implication	4
$\iff$	Biconditional or Equivalence	5

3.206 Laws of propositional logic

The following table summarizes the laws of Propositional Logic.

	Disjunction	Conjunction
Idempotent laws	$p \lor p \equiv p$	$p \land p \equiv p$
Commutative laws	$p \lor q \equiv q \lor p$	$p \land q \equiv q \land p$
Associative laws	$(p \lor q) \lor r \equiv p \lor (q \lor r)$	$(p \land q) \land r \equiv p \land (q \land r)$
Distributive laws	$p \lor (q \land r) \equiv (p \lor q) \land (p \lor r)$	$p \land (q \lor r) \equiv (p \land q) \lor (p \land r)$
Identity laws	$p \lor \mathbf{F} \equiv p$	$p \land \mathbf{T} \equiv p$
Domination laws	$p \lor \mathbf{T} \equiv \mathbf{T}$	$p \land \mathbf{F} \equiv \mathbf{F}$
De Morgan's laws	$\neg(p \lor q) \equiv \neg p \land \neg q$	$\neg(p \land q) \equiv \neg p \lor \neg q$
Absorption laws	$p \lor (p \land q) \equiv p$	$p \land (p \lor q) \equiv p$
Negation laws	$p \lor \neg p \equiv \mathbf{T}$[1]	$p \land \neg p \equiv \mathbf{F}$[2]
Double negation law	$\neg\neg p \equiv p$	

[1] A statement that's always **true** is a *Tautology*

[2] A statement that's always **false** is a *Contradiction*

Week 7

Learning Objectives

- Describe the basic concepts of predicate logic.
- Describe existential and universal quantifiers.
- Assign truth values to quantified statements.

4.101 Introduction to predicate logic

Our previous Propositional Logic is useful for studying propositions but has some limitations:

- It cannot express precisely the meaning of complex mathematical statements.
- It only studies propositions, i.e. statements with known truth values

Predicate Logic is a different type of Mathematical Logic which overcomes the limitations of Propositional Logic and can be used to build more complex reasoning.

Example 1

Given the statements:

- *All **men** are **mortal**.*
- ***Socrates** is a **man**.*

It's natural to conclude that *Socrates* is a *man*. This sort of reasoning cannot be expressed by Propositional Logic. Predicate Logic enables us to formalise it.

Example 2

Given the statement ***x** square is equal to 4*. We **know** this statement is **NOT** a proposition as its truth value is a function depending on **x**, however Predicate Logic can express and formalise this statement.

4.103 What are predicates?

We start with some examples of statements which cannot be expressed by Propositional Logic.

Insufficiency of Propositional Logic

Going back to the previous example ***x** squared is equal to 4*. We already saw that this statement cannot be a proposition because its truth value is a function depending on **x**.

Definition of Predicate

Predicates are generalizations of propositions. They are (Boolean) functions which return *TRUE* or *FALSE* depending on their variables. They **become** propositions when their variables are assigned values.

Predicates, much like regular sentences, are composed of smaller parts. The statement ***x** square is equal to 4* contains two parts: the variable **x**; and the predicate **is equal to 4**.

We can formalize this statement as $P(x)$ where P is the predicate *squared is equal to 4* and x is the variable.

P is referred to as the *Propositional Function*.

As soon as a value is assigned to the variable x, the statement $P(x)$ becomes a proposition and has a truth value.

Example 1 Let x be an integer and let P be the propositional function *square is equal to 4*, therefore $P(2)$ is *TRUE* and $P(3)$ is *FALSE*.

Predicates with multiple variables

It's important to note that Predicates can depend on **more than one** variable.

Example 1 Let $P(x, y)$ denote $x^2 > y$, therefore $P(-2, 3) \equiv 4 > 3$ is *TRUE* and $P(2, 4) \equiv 4 > 4$ is *FALSE*.

Example 2 Let $Q(x, y, z)$ denote $x + y < z$, therefore $Q(2, 4, 5) \equiv 2 + 4 < 5$ which is *FALSE*, $Q(1, 2, 4) \equiv 1 + 2 < 4$ which is *TRUE*, and $Q(1, 2, z)$ is **NOT** a proposition.

Logical operations

All logic previously defined for propositional logic carries over to predicate logic.

Example 1 If $P(x)$ denotes $x^2 < 16$, then $P(1) \vee P(-5) \equiv (1 < 16) \vee (25 < 16) \equiv \mathbf{T} \vee \mathbf{F} \equiv \mathbf{T}$.

4.105 Quantification

Quantification expresses the extent to which a predicate is true over a range of elements.

The two most most important quantifiers are the universal quantifier $\forall$ and the existential quantifier $\exists$.

There is a third quantifier called the uniqueness quantifier $\exists!$.

Example 1 The following statements give examples of **quantified predicates**.

- *All men are mortal.*
- *Some computers are not connected to the network.*

Universal Quantifier $\forall$

The Universal Quantification of a predicate $P(x)$ is the proposition:

- $P(x)$ is true for all values of x in the universe of discourse.

We use the notation $\forall x P(x)$ and read it as *for all x*.

If the universe of discourse is finite $\{n_1, n_2, \ldots, n_k\}$ then the universal quantifier is the **conjunction** of the propositions over all elements: $\forall x P(x) \equiv P(n_1) \wedge P(n_2) \wedge \ldots \wedge P(n_k)$.

Example 1 Let P, Q denote the following propositional functions of x:

- $P(X)$: x must take a discrete mathematics course
- $Q(X)$: x is a Computer Science student

Where the universe of discourse for both $P(x)$ and $Q(x)$ is all university students. Let's express the following statements symbolically:

- Every CS student must take a course on discrete mathematics.

  $\forall x Q(x) \implies P(x)$
- Everybody must take a discrete mathematics course or be a CS student.

  $\forall x (P(x) \vee Q(x))$
- Everybody must take a discrete mathematics course and be a CS student.

  $\forall x (P(x) \wedge Q(x))$

Example 2 Let's formalise the following statement S

- S: For every x and for every y, $x + y > 10$

  Let $P(x, y)$ be the statement $x + y > 10$, where the universe of discourse is the set of all integers.

  $\forall x \forall y P(x, y) \equiv \forall x, y P(x, y)$

Existential Quantifier $\exists$

The existential quantification of a predicate $P(x)$ is the proposition *There exists a value of x in the universe of discourse, such that $P(x)$ is true.*

If the universe of discourse is finite $\{n_1, n_2, \ldots, n_k\}$, then the existential quantifier is the **disjunction** of the proposition over all elements: $\exists x P(x) \equiv P(n_1) \vee P(n_2) \vee \ldots \vee P(n_k)$.

Example 1 Let $P(x, y)$ denote the statement $x + y = 5$. The expression $\exists x \exists y P(x, y)$ means "There exists a value x and a value y in the universe of discourse such that $x + y = 5$ is true".

Example 2 Let a, b, c denote fixed real numbers and S be the statement /There exists a real solution to $ax^2 + bx - c = 0$. S can be expressed as $\exists x P(x)$.

Uniqueness Quantifier $\exists!$

The uniqueness quantification of a predicate $P(x)$ is the proposition *There exists a unique value x in the universe of discourse such that $P(x)$ is true.*

The uniqueness quantifier is a **special case** of the existential quantifier.

We use the notation $\exists! x P(x)$ and read it as /there exists a unique x".

Example 1

Let $P(x)$ be the statement $x^2 = 4$. The expression $\exists! x P(x)$ means "There exists a unique value of x such that $x^2 = 4$ is true".

4.107 Nested Quantifiers

When we want to express statements with multiple variables, we employ nested quantifiers.

Nested Quantifier	Meaning
$\forall x \forall y P(x, y)$	$P(x, y)$ is true for every pair (x, y)
$\exists x \exists y P(x, y)$	There is a pair (x, y) for which $P(x,y) is true
$\forall x \exists y P(x, y)$	For all x, there is a y for which P(x,y) is true
$\exists x \forall y P(x, y)$	There is an x for which $P(x, y)$ is true for all y

Binding Variables

A variable is said to be **bound** if it is within the scope of a quantifier. A variable that is **not bound** is called a **free** variable.

Example 1 Let $P be a propositional function and S be the statement $\exists x P(x, y)$. In this case, x is **bound** while y is **free**.

Logical operations

All the logical operations discussed previously, can also be applied to quantified statements.

Order of operations

When we have quantifiers of the same type, either all universal or all existential, the other doesn't matter. However, when we're dealing with quantifiers of different types we **must** apply the quantifiers at the correct order.

Example 1 $\forall x \forall y P(x, y) \equiv \forall y \forall x P(x, y)$

However

$\forall x \exists y P(x, y) \neq \exists y \forall x P(x, y)$

Precedence of Quantifiers

The quantifiers $\forall$ and $\exists$ have precedence over **all** other logical operators. This means that $\forall x P(x) \vee Q(x)$ should be read as $(\forall x P(x)) \vee Q(x)$ and $\forall x P(x) \implies Q(x)$ is to be read as $(\forall x P(x)) \implies Q(x)$.