Modelling Computing Systems 1

Lecturer: Faron Moller

Lectures: Three hours per week

Problem Sessions: One hour per week

Assessment: 30% continuous assessment (weekly)
70% examination (January)

- You will complete a worksheet in every weekly problem session, of which there will be ten in total.
- Each worksheet is worth 3%, for a total of 30% of your module mark.
- You **must** attend your weekly problem session to do these worksheets; each one you miss loses you 3%.
- Each worksheet will be based on the material covered in the lectures during the previous week.

Overview of the Module

CS-170 & CS-175 (Modelling Computing Systems 1 & 2) will follow, *very closely*, the following textbook:

> F Moller & G Struth, **Modelling Computing Systems**, Springer 2013.

In particular, CS-170 will cover the following five chapters (with roughly two weeks spent on each chapter):

- **Chapter 1: Propositional Logic**
  reasoning about the truth of statements
- **Chapter 2: Sets**
  collections of data
- **Chapter 4: Predicate Logic**
  combining sets and logic
- **Chapter 6: Functions**
  transforming data
- **Chapter 7: Relations**
  relationships between data

There are many other textbooks (in the library) which cover the above topics; pretty much anything with the phrase *"Discrete Mathematics"* in its title will do.

Modelling – What It's All About

From *docs.microsoft.com/en-us/legal/termsofuse*:

"MICROSOFT CORPORATION HEREBY DISCLAIMS ALL WARRANTIES AND CONDITIONS WITH REGARD TO THE SOFTWARE, INCLUDING ALL WARRANTIES AND CONDITIONS OF MERCHANTABILITY, WHETHER EXPRESS, IMPLIED OR STATUTORY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT.
...
IN NO EVENT SHALL MICROSOFT AND/OR ITS RESPECTIVE SUPPLIERS BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF SOFTWARE, DOCUMENTS, PROVISION OF OR FAILURE TO PROVIDE SERVICES, OR INFORMATION AVAILABLE FROM THE SERVICES. "

(Though it appears in the small print that no one reads, this text is written by them in upper case – ie, digital shouting.)

Discuss!

A Few Case Studies

USS Scorpion

P G Neumann, *Computer Related Risks*, Addison Wesley, 1994.

Therac 25 Radiotherapy Machine

N. Leveson and N., C.S. Turner, "An Investigation of the Therac-25 Accidents," *IEEE Computer* **26**(7):18–41, July 1993.

The Intel Pentium Bug

T.R. Halfhill, "The Truth Behind the Pentium Bug," *Byte*, March 1995.

Ariane 5

http://www.esa.int/export/esaCP/Pr_33_1996_p_EN.html.

A Few More Case Studies

London Ambulance Service

A. Finkelstein and J. Dowell, "A Comedy of Errors: The London Ambulance Service Case Study," in *The Eighth International Workshop on Software Specification and Design*, IEEE CS Press, pp2–4, 1996.

Clayton Tunnel Accident

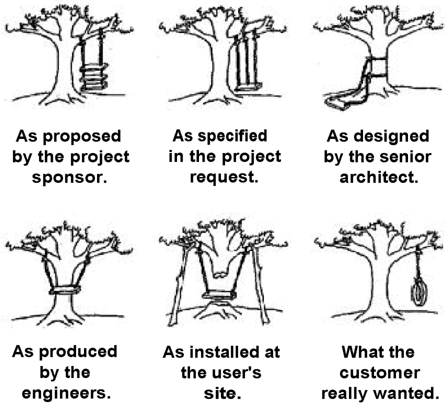
L.T.C. Rolt, *Red for Danger: The Classic History of Railway Disasters*, The History Press, 2009.

Needham-Schroeder Protocol

G. Lowe, "An attack on the Needham-Schroeder public key authentication protocol," *Information Processing Letters* **56**(3):131-136, November 1995.

Common Themes Behind Failures

- **Requirements Engineering** Specifying exactly how a software system should behave can be very difficult.
- **Unintended Consequences** The implications of design decision are rarely understood or appreciated.



A Few Definitions

model.

1. A miniature representation of a thing, with the several parts in due proportion; sometimes, a facsimile of the same size.
2. Something intended to serve, or that may serve, as a pattern of something to be made; a material representation of or embodiment of an ideal; sometimes a drawing or a plan; a description of observed behaviour, simplified by ignoring certain details.

abstraction.

The act or process of leaving out of consideration one or more properties of a complex object so as to attend to others.

A Few More Definitions

system.

An assemblage of objects arranged in a regular subordination, or after some distinct method, usually logicial or scientific; a complete whole of objects related by some common law, principle, or end; a complete exhibition of essential principles or facts, arranged in a rational dependence or connection; a regular union of principles or parts forming one entire thing.

notation.

Any particular system of characters, symbols, or abbreviated expressions used in art, or in science, to express briefly technical facts, quantities, etc. Especially the system of figures, letters, and signs used in arithmetic and algebra to express number, quantity, or operations.

"By relieving the mind of all unnecessary work, a good notation sets it free to concentrate upon more advanced problems, and in effect increases the mental power of the race."

Alfred North Whitehead.

Propositions Logic

Consider the following argument, made by a person taking the pulse of an unconscious man:

1. *Either this man is dead or my watch has stopped.*
2. *My watch is still ticking.*
3. *Therefore, this man is dead.*

Here, we ***deduce***, or ***infer***, a ***conclusion*** – a particular ***proposition***, or ***statement*** (that a man is dead) – based on the truth of two other propositions (***premises***).

Of course this argument is not valid. Such failures in reasoning underpin software system failures.

The role of ***Propositional Logic*** is to formalise such reasoning to ensure only valid arguments can be made.

Such arguments have been studied for millennia; here is a valid argument made by Aristotle (384-322 BC):

1. *All men are mortal.*
2. *Socrates is a man.*
3. *Therefore, Socrates is mortal.*

Propositions / Statements

Propositions, or ***statements***, are declarations which are either *true* or *false*.

Exercise: Which of the following are propositions ?

1. $2+3=5$.
2. $2+3=6$.
3. $2+x=6$.
4. Do your homework, Joel!
5. Joel didn't do his homework.
6. Is there life on Mars?
7. False
8. What Felix says is false.
9. What this sentence says is false.

Answer:

- 1, 2, 5, 7 and 8 are propositions;
- 3, 4, 6 and 9 are *not* propositions.

Deductions / Inferences

A ***deduction***, or ***inference***, is when you deduce, or infer, one statement (a ***conclusion***) from another statement (a ***premise***).

Such a deduction can be logically ***valid*** or logically ***invalid***.

Exercise: Which of the following are valid deductions?

1. All men have green blood.

   Socrates is a man.

   Therefore, Socrates has green blood

2. If the fire alarm sounds,
   then everyone must leave the building.

   Everyone is leaving the building.

   Therefore, the fire alarm has sounded.

3. If the fire alarm sounds,
   then everyone must leave the building.

   The fire alarm has sounded.

   Therefore, everyone is leaving the building.

The Language of Propositional Logic

Statements (i.e., formulæ) of Propositional Logic are similar to algebraic expressions, like $x^2+2y-15$.

- Algebraic expressions evaluate to *numerical values* based on the values of the variables in the expression.
- Propositional formulæ evaluate to *truth values* based on the values of the variables in the statement.

For ***propositional variables*** we will always use upper-case letters or (more commonly) meaningful words starting with an upper-case letter.

Example: Let Dead represent the statement:

"This man is dead".

Expressions in algebra are build using addition ($+$), subtraction ($-$), multiplication ($\times$), division ($\div$).

Propositional logic uses five ***propositional connectives***: *not* ($\neg$); *or* ($\vee$); *and* ($\wedge$); *implies* ($\Rightarrow$); *if, and only if* ($\Leftrightarrow$).

Negation

connective: negation
symbol: $\neg p$
pronounced: "not p"

In English:

- *not p (or, rather, statement p with "not" after verb)*
- *p does not hold / is not true / is false*
- *it is not the case that p*

Example:

If Dead stands for "This man is dead," then $\neg\mathrm{Dead}$ says

- "This man is not dead"
- "It is not the case that this man is dead"

Observation: If a proposition is not true, then it must be false. Conversely, if it is not false, then it must be true.
In particular, if a proposition is *not* not true, then it is true:

$\neg\neg p$ is the same as p (*Law of Double Negation*)

Exercise: Rewrite the following statements without negations at the start.

1. $\neg$ "The Earth revolves around the sun."
2. $\neg$ "All of my children are boys."
3. $\neg(2 + 2 \leq 4)$.

Disjunction

connective: disjunction
symbol: $p \vee q$ (p and q are called *disjuncts*)
pronounced: "p or q"
in English:

- p *or* q
- p *or* q *or both*
- p *and/or* q
- p *unless* q

Example: If Dead stands for "This man is dead," and Watch for "My watch has stopped" then $\mathrm{Dead} \vee \mathrm{Watch}$ says

- "Either this man is dead or my watch has stopped"
- "If this man is alive, then my watch must have stopped"

(Possibly the man is dead *and* the watch has stopped.)

Recall: $\neg p$ is true *if* p is *not* true, thus

$p \vee \neg p$ must always be true. (*Law of Excluded Middle*)

Disjunction

Exercise: Are the following disjunctions true or false?

1. $(3 < 2) \vee (3 < 5)$
2. $(5 < 4) \vee (7 < 5)$
3. $(5 < 6) \vee (6 < 8)$

Note: We use $\vee$ for the *inclusive or*: $p \vee q$ is true if (though not only if) *both* p and q are true.

Sometimes an *exclusive* interpretation is intended:

Either you be quiet or you won't get an ice cream.

This kind of disjunction, written $\oplus$, is called *exclusive or.*

Exercise: For each of the following disjuctive statements, decide whether you think the speaker intends to use the inclusive or exclusive sense of disjunction.

1. Joel came in last place in the round-robin competition; thus, either Felix beat him or Oskar beat him.
2. The light is either on or off.
3. You can have tea or coffee.

Conjunction

connective: conjunction
symbol: $p \wedge q$ (p and q are called *conjuncts*)
pronounced: "p and q"
in English:

- p *and* q;
- p *but* q;
- *not only* p *but also* q.

Example: If Dead stands for "This man is dead," and Watch for "My watch has stopped" then $\mathrm{Dead} \wedge \mathrm{Watch}$ says

- "This man is dead *and* my watch has stopped"
- "Not only is this man dead, but so is my watch"

Recall: $\neg p$ is false if p is true, thus

$p \wedge \neg p$ must always be false.

Exercise: Are the following conjunctions true or false?

1. $(3 < 2) \wedge (3 < 5)$
2. $(5 < 4) \wedge (7 < 5)$
3. $(5 < 6) \wedge (6 < 8)$

Implication

connective: implication
symbol: $p \Rightarrow q$
pronounced: "p implies q"
in English:

- p *implies* q
- *if* p *then* q
- q *if* p
- p *only if* q
- q *whenever* p
- p *is a sufficient condition for* q
- q *is a necessary condition for* p

Example: If $\mathrm{SignalRed}$ stands for "This signal shows red" and $\mathrm{TrainStop}$ stands for "The train stops", then $\mathrm{SignalRed} \Rightarrow \mathrm{TrainStop}$ says

"If the signal shows red then the train stops".

The only scenario in which this statement is false is if the signal shows red but the train does not stop.

Hence the statement does not contradict the situation in which the signal does not show red and yet the train nevertheless stops.

Implication

Exercise: Let $\mathrm{JoelHappy}$ stand for "Joel is happy," and let $\mathrm{AmandaHappy}$ stand for "Amanda is happy." Each of the three statements below translates as either

$$\mathrm{JoelHappy} \Rightarrow \mathrm{AmandaHappy}$$

or as

$$\mathrm{AmandaHappy} \Rightarrow \mathrm{JoelHappy}$$

Determine which in each case.

1. "Joel is happy whenever Amanda is happy."
2. "Joel is happy only if Amanda is happy."
3. "Joel is happy unless Amanda is not happy."

Exercise: On the door of a particular house is the following warning to potential thieves:

Barking dogs don't bite.

My dog doesn't bark.

Should a potential thief *necessarily* be concerned?

Equivalence

connective: equivalence
symbol: $p \Leftrightarrow q$
pronounced: "p if, and only if, q"
in English:

- p *if, and only if,* q
- p *is equivalent to* q
- p *is a necessary and sufficient condition for* q

Example: Let

- $\mathrm{TrainEnter}$ stand for "The train enters the tunnel" and
- $\mathrm{TunnelClear}$ stand for "The tunnel is clear."

Then $\mathrm{TrainEnter} \Leftrightarrow \mathrm{TunnelClear}$ says

"The train enters the tunnel
if, and only if, the tunnel is clear."

This statement is false if one of these two scenarios holds:

- the train enters the tunnel while the tunnel is not clear;
- the tunnel is clear but the train does not enter.

The Syntax of Propositional Logic

Definition A ***propositional formula*** is either

- an ***atomic formula*** typically a variable P, Q, R etc.; or
- a ***compound formula*** build up using operators.

Two special atomic formulæ:

- ***true*** (the proposition which is always true)
- ***false*** (the proposition which is always false)

In summary: If p and q are propositional formulæ, then so are the following:

true		*truth*
false		*falsity*
P		*variable*
$\neg p$	*not* p	*negation*
$p \vee q$	p *or* q	*disjunction*
$p \wedge q$	p *and* q	*conjunction*
$p \Rightarrow q$	p *implies* q	*implication*
$p \Leftrightarrow q$	p *if, and only if,* q	*equivalence*

A formula built using these rules is called *a well-formed formula (wff).*

Note: p, q,... are *not* propositional variables; rather, they are *metavariables* which stand for arbitrary formulæ.

Parentheses and Precedences

We use parentheses to disambiguate expressions.

For example, $P \vee Q \Rightarrow R$ could be read:

- either as $(P \vee Q) \Rightarrow R$
- or as $P \vee (Q \Rightarrow R)$.

Thus, we extend the definition of well-formed formulæ to include parentheses.

We also define a precedence for connectives to reduce the need for parentheses (and thus increased readability without creating ambiguity):

- $\neg$ binds more tightly than $\wedge$ and $\vee$
- $\wedge$ and $\vee$ bind more tightly than $\Rightarrow$ and $\Leftrightarrow$

Apart from this, connectives will be applied right-to-left.

Example: $P \Rightarrow Q \wedge R \Rightarrow S$ reads as $P \Rightarrow ((Q \wedge R) \Rightarrow S)$

Defining Further Propositional Operators

The "exclusive or" operation can be expressed, by noting that $p \oplus q$ says that one of p and q is true if, and only if, the other one is *not* true:

$$p \oplus q \quad = \quad p \Leftrightarrow \neg q$$

or, equivalently, by

$$p \oplus q \quad = \quad \neg p \Leftrightarrow q$$

The tempting definition

$$p \oplus q \quad = \quad (p \Leftrightarrow \neg q) \wedge (\neg p \Leftrightarrow q)$$

would also be correct, but would be an overkill: with a little thought we realise that $p \Leftrightarrow \neg q$ is the same as $\neg p \Leftrightarrow q$.

Exercise: Express the following connectives using the connectives of propositional logic.

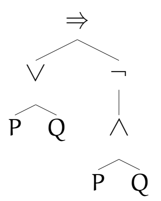
1. The NAND connective $p \mid q$ which is true if, and only if, p and q are not both true.

2. The NOR connective $p \downarrow q$ which is true if, and only if, neither p nor q are true.

Syntax Trees

Well-formed formulæ can be viewed as syntax trees:

$(P \vee Q) \Rightarrow \neg(P \wedge Q)$ corresponds to:

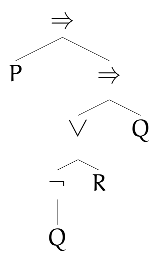


The process of recognising $(P \vee Q) \Rightarrow \neg(P \wedge Q)$ as a well-formed formula is directly reflected in the syntax tree.

The syntax tree makes parsing clear without the need for parentheses or precedence rules. Without rules of precedence, $P \vee Q \Rightarrow \neg P \wedge Q$ can be read in many different ways, all having different meanings and syntax trees.

Example: $P \Rightarrow \neg Q \vee R \Rightarrow Q$

corresponds to the following syntax tree according to precedence rules:



The syntax tree also helps to evaluate this expression based on truth values for variables P, Q, R, or to determine how to place brackets.

(Non-)Well-formed Formulæ (wff)

Example: The string of symbols

$$\neg(P \wedge (Q \vee \neg))$$

is *not* a well-formed formula, which can be seen by applying formation rules for propositional formulæ backwards (reflected also in the attempt to draw its syntax tree.)

Exercise: Which of the following are wff? Rewrite each wff using a minimal number of parentheses without changing its meaning, and draw its syntax tree.

1. $((P \Rightarrow Q) \Leftrightarrow (Q \Rightarrow P))$
2. $P \vee Q(\wedge P)$
3. $(P \vee Q) \wedge P$

Modelling with Propositional Logic

Propositional Logic is very important for modelling real-life scenarios, in which propositional variables represent particular properties which can be true or false.

Example: Consider the following lines of code:

```
...
if CabinPressure < MinPressure
  then PrepareForLanding;
if FlightHeight < MinHeight
  then  PrepareForLanding;
...
```

A software engineer optimises this to

```
...
if (CabinPressure < MinPressure
      and FlightHeight < MinHeight)
  then  PrepareForLanding;
..
```


Modelling with Propositional Logic

Using variables

- $\mathrm{Pressure}$ for `CabinPressure < MinPressure`
- Height for `FlightHeight < MinHeight`
- Land for `PrepareForLanding`

we can express the program as a propositional formula

$$(\mathrm{Pressure} \Rightarrow \mathrm{Land}) \ \wedge \ (\mathrm{Height} \Rightarrow \mathrm{Land})$$

while the suggested optimisation corresponds to

$$(\mathrm{Pressure} \wedge \mathrm{Height}) \ \Rightarrow \ \mathrm{Land}$$

Considering when these two formulæ are false, we see that they are *not* equivalent. The correct variant, one which is equivalent to the formula corresponding to the program, would be

$$(\mathrm{Pressure} \vee \mathrm{Height}) \Rightarrow \mathrm{Land}$$

That is, the optimised code should be

```
...
if (CabinPressure < MinPressure
      or FlightHeight < MinHeight)
  then  PrepareForLanding;
..
```


Modelling with Propositional Logic

Example: Consider following four symbols:

○ ● □ ■

Define the following propositional variables:

- B stands for *The symbol in question is black.*
- C stands for *The symbol in question is a circle.*

The following table stores the truth of some propositional formulæ w.r.t. one of the above mentioned four symbols.

	○	●	□	■	
B	×	✓	×	✓	(it's black)
$\neg\mathrm{B}$	✓	×	✓	×	(it's **not** black)
$\mathrm{B} \vee \mathrm{C}$	✓	✓	×	✓	(it's black **or** it's a circle)
$\mathrm{B} \wedge \mathrm{C}$	×	✓	×	×	(it's black **and** it's a circle)
$\mathrm{B} \Rightarrow \mathrm{C}$	✓	✓	✓	×	(**if** it's black **then** it's a circle)
$\mathrm{B} \Leftrightarrow \mathrm{C}$	×	✓	✓	×	(it's black **if and only if** it's a circle)

Ambiguities of Natural Languages

We have already seen many ambiguous sentences formulated in English. Here are a few more examples.

Example: Consider the following statement, made by a father to his unruly children during long drives in the car:

Everyone who sits quietly for the next hour will get an ice cream.

What exactly does this statement say? More importantly, does it express what the father means to say? You may imagine that he wants to suggest that

Anyone who misbehaves will <u>not</u> get an ice cream.

But this does *not* follow from his statement: the children who get ice cream will include those who sit quietly, but may well include the noisy ones as well.

The father made this statement to manipulate language to his benefit: He was being intentionally vague, relying on his children to misinterpret his statement as saying something more than it actually does, namely that any misbehaving children will not get ice cream.

A Riddle

Exercise: Consider the following four symbols:

○ ● □ ■

I have in mind one of these four symbols. I will *accept* any symbol which either has the same colour *or* the same shape (or both) as the one I have in mind, otherwise I will *reject* it.

If I accept the black square, what does this suggest to you about whether I accept or reject the other three symbols?

The answer may be tricky to realise. But at the heart of it is the type of unintended consequences which permeate our computational problem-solving thinking.

Truth Tables

We will now *define* the meaning of the propositional connectives. In doing this, the *semantics* of propositional logic will be formally, rigorously and unambiguously defined.

For each connective we will explicitly list its truth value depending on all possible combination of truth values for its constituent propositions.

Negation:

p	$\neg p$
F	**T**
T	**F**

Disjunction and Conjunction:

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

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

Implication and Equivalence:

p	q	$p \Rightarrow q$
F	**F**	**T**
F	**T**	**T**
T	**F**	**F**
T	**T**	**T**

p	q	$p \Leftrightarrow q$
F	**F**	**T**
F	**T**	**F**
T	**F**	**F**
T	**T**	**T**

Truth Tables

Example: Consider the following statement:

Everyone who sits quietly for the next hour will get an ice cream.

This can be formalised as follows. Let

- Quiet stand for *You sit quietly*.
- Ice stands for *You get an ice cream*.

Then the statement translates to $\mathrm{Quiet} \Rightarrow \mathrm{Ice}$ which has the following truth table:

Quiet	Ice	$\mathrm{Quiet} \Rightarrow \mathrm{Ice}$
F	**F**	**T**
F	**T**	**T**
T	**F**	**F**
T	**T**	**T**

The *only* scenario in which the above statement can be considered false is:

You do not get an ice cream despite being quiet.

In particular, in the case where a noisy child gets an ice cream, the statement still is true (second row of truth table)

Truth Tables

Example: Catherine wishes to go to a party with either Jim or Jules. She is currently dating both, thus she doesn't want to go if they will both be there.

Let
- Cat stand for *Catherine goes to the party*
- Jim stand for *Jim goes to the party*
- Jules stand for *Jules goes to the party*

Catherine's predicament then can be formalised as

$$\mathrm{Cat} \Rightarrow \neg(\mathrm{Jim} \wedge \mathrm{Jules})$$

That is, Catherine goes to the party *only if* Jim and Jules don't both go to the party. It has the following truth table:

Cat	Jim	Jules	$\mathrm{Jim} \wedge \mathrm{Jules}$	$\neg(\mathrm{Jim} \wedge \mathrm{Jules})$	$\mathrm{Cat} \Rightarrow \neg(\mathrm{Jim} \wedge \mathrm{Jules})$
F	**F**	**F**	**F**	**T**	**T**
F	**F**	**T**	**F**	**T**	**T**
F	**T**	**F**	**F**	**T**	**T**
F	**T**	**T**	**T**	**F**	**T**
T	**F**	**F**	**F**	**T**	**T**
T	**F**	**T**	**F**	**T**	**T**
T	**T**	**F**	**F**	**T**	**T**
T	**T**	**T**	**T**	**F**	**F**

This tells us that the proposition is *false* if, and only if, all three participants in this love triangle go to the party.

Truth Tables

A more succinct way of filling the truth table is as follows:

Cat	Jim	Jules	Cat	$\Rightarrow$	$\neg$(Jim		$\wedge$	Jules)
F	**F**	**F**	**F**	**T**	**T**	**F**	**F**	**F**
F	**F**	**T**	**F**	**T**	**T**	**F**	**F**	**T**
F	**T**	**F**	**F**	**T**	**T**	**T**	**F**	**F**
F	**T**	**T**	**F**	**T**	**F**	**T**	**T**	**T**
T	**F**	**F**	**T**	**T**	**T**	**F**	**F**	**F**
T	**F**	**T**	**T**	**T**	**T**	**F**	**F**	**T**
T	**T**	**F**	**T**	**T**	**T**	**T**	**F**	**F**
T	**T**	**T**	**T**	**F**	**F**	**T**	**T**	**T**
(0)	(0)	(0)	(1)	**(4)**	(3)	(1)	(2)	(1)

The numbering at the bottom is just to explain the table, and indicates the order in which the columns of entries are added to the table:

[0]

The variables columns are inserted first

(1) The variables columns are copied

(2) The most tightly-binding operators come next

(3) Further operator columns are filled in

(4) The outermost operator column is inserted last

Truth Tables

Exercise: Construct the truth table for $\neg(P \Leftrightarrow \neg Q)$

P	Q	$\neg$	(	P	$\Leftrightarrow$	$\neg$	Q	)
F	**F**							
F	**T**							
T	**F**							
T	**T**							

Exercise: How many rows will there be in a truth table involving four propositional variables P, Q, R and S?

What if there are five propositional variables?

What if there are n propositional variables?

Equivalences and Valid Arguments

Propositions can be expressed in propositional logic in different yet equivalent ways. For example,

$\text{Cat} \Rightarrow \neg(\text{Jim} \wedge \text{Jules})$ *"If Cat then not both of Jim and Jules."*

is equivalent to the following

$\neg(\text{Cat} \wedge \text{Jim} \wedge \text{Jules})$ *"Cat, Jim and Jules cannot all be true."*

$\neg\text{Cat} \vee \neg\text{Jim} \vee \neg\text{Jules}$ *"One of Cat, Jim or Jules is false."*

Ways to verify that two formulæ p and q are equivalent:

1. Construct truth tables for p and q, and observe that they have the same truth values under all interpretations of their respective atomic propositions.
2. Build a truth table for $p \Leftrightarrow q$, and observe that it is true under all interpretations.

If so, p and q are said to be ***logically equivalent.***

Let p be a formula.

1. p is a ***tautology*** if, and only if, it is true under all interpretations. In this case we say p is ***valid***.
2. p is a ***contradiction*** if, and only if, it is false under all interpretations. In this case we say p is ***unsatisfiable***.
3. p is ***satisfiable*** if, and only if, it is true under some interpretations; i.e., it is *not* a contradiction.

Equivalences and Valid Arguments

Example:

- $p \vee \neg p$ is a tautology and
- $p \wedge \neg p$ is a contradiction

which can be formally verified by constructing their truth tables:

p	$\neg p$	$p \vee \neg p$	$p \wedge \neg p$
F	**T**	**T**	**F**
T	**F**	**T**	**F**

Exercise: Construct truth tables for each of the following formulæ to determine which are tautologies and which are contradictions.

1. $p \vee (\neg p \wedge q)$
2. $(p \wedge q) \wedge \neg(p \vee q)$
3. $(p \Rightarrow \neg p) \Leftrightarrow \neg p$

Equivalences and Valid Arguments

Tautologies are important in ascertaining the validity of arguments. Consider our first argument:

1. Either this man is dead or my watch has stopped.
2. My watch is still ticking.
   Therefore
3. This man is dead.

This argument is valid if the conjunction of the two premises implies the conclusion, that is, if

$$(\mathrm{Dead} \vee \mathrm{Watch}) \wedge \neg\mathrm{Watch} \Rightarrow \mathrm{Dead}$$

is a tautology. We can confirm this by constructing its truth table:

Dead	Watch	(Dead	$\vee$	Watch)	$\wedge$	$\neg$	Watch	$\Rightarrow$	Dead
F	F	F	F	F	F	T	F	**T**	F
F	T	F	T	T	F	F	T	**T**	F
T	F	T	T	F	T	T	F	**T**	T
T	T	T	T	T	F	F	T	**T**	T
(0)	(0)	(1)	(2)	(1)	(3)	(2)	(1)	**(4)**	(1)

Equivalences and Valid Arguments

In contrast, consider the following argument:

1. If my dog barks, then my dog doesn't bite.
2. My dog doesn't bark.
   Therefore
3. My dog bites.

Its formalisation yields the following truth table:

Barks	Bites	(Barks	$\Rightarrow$	$\neg$	Bites)	$\wedge$	$\neg$	Barks	$\Rightarrow$	Bites
F	F	F	T	T	F	T	T	F	**F**	F
F	T	F	T	F	T	T	T	F	**T**	T
T	F	T	T	T	F	F	F	T	**T**	F
T	T	T	F	F	T	F	F	T	**T**	T
(0)	(0)	(1)	(3)	(2)	(1)	(4)	(2)	(1)	**(5)**	(1)

The first row of this truth table shows that the proposition – and hence the argument it represents – is not valid.

That is, it presents a scenario in which the proposition may be false: we may have a dog that neither barks nor bites.

Equivalences and Valid Arguments

Exercise: We represented a piece of computer program above in propositional logic as:

$$p \quad = \quad (\mathrm{Pressure} \Rightarrow \mathrm{Land}) \wedge (\mathrm{Height} \Rightarrow \mathrm{Land})$$

We also considered two optimisations of this program represented as

$$q \quad = \quad \mathrm{Pressure} \wedge \mathrm{Height} \Rightarrow \mathrm{Land}$$
$$r \quad = \quad \mathrm{Pressure} \vee \mathrm{Height} \Rightarrow \mathrm{Land}$$

Of course, an optimisation is only correct if the representation of the optimised program is equivalent to the original one.

By constructing appropriate truth tables, explain which of the two optimisations is correct and which is not.

Algebraic Laws for Logical Equivalences

Using truth tables to prove equivalences between propositions can quickly become tedious. However, we can avoid relying on truth tables by reasoning equationally much as we do in algebra and arithmetic. Once we have determined that two propositions p and q are equivalent, we can replace one with the other. A large number of such equivalence are know, which we can use as "algebraic laws". Here we list a few of them:

Commutative Laws

$$p \vee q \;\Leftrightarrow\; q \vee p \qquad p \wedge q \;\Leftrightarrow\; q \wedge p$$

Double Negation Law

$$\neg\neg p \;\Leftrightarrow\; p$$

Implication Law

$$p \Rightarrow q \;\Leftrightarrow\; \neg p \vee q$$

Implication Law

$$p \Rightarrow q \;\Leftrightarrow\; \neg p \vee q$$

E.g., we can verify the **Contrapositive Law** as follows:

$$\begin{array}{rcll} p \Rightarrow q & \Leftrightarrow & \neg p \vee q & \textit{(Implication)} \\ & \Leftrightarrow & q \vee \neg p & \textit{(Commutativity)} \\ & \Leftrightarrow & \neg\neg q \vee \neg p & \textit{(Double Negation)} \\ & \Leftrightarrow & \neg q \Rightarrow \neg p & \textit{(Implication)} \end{array}$$

Sets

Propositional logic is about reasoning, and inferring new facts from known facts.

But we also need to model different sorts of data in order to reason about it.

The most basic data structure is a ***set***, which is simply a collection of objects.

These objects typically share a property. For example:

- The set of students enrolled in CS-170.
- The set of (named) beaches on the Gower Peninsula.
- The set of programs in a particular computer library.

The objects of a set are referred to as its ***elements***, or ***members***.

The ***cardinality*** (or ***size***) of a set A, denoted by $|A|$, is the number of objects in A.

Set Notation

Sets are depicted by listing their elements, separated by commas, within curly braces. For example:

- $\{3, 7, 13\}$
- $\{\text{red}, \text{blue}, \text{yellow}\}$.
- $\{\text{Joel}, \text{Felix}, \text{Oskar}, \text{Amanda}, \text{Maya}, \text{Vera}, \text{Jonathan}\}$
- $\{1, 3, 5, \ldots, 99\}$
- $\{a, b, c, \ldots, z\}$
- $\{2, 3, 5, 7, 11, 13, 17, \ldots\}$

The last example (and the two before it) may be ambiguous. To avoid this, we use explicit ***set building notation***:

$$\{ x : x \text{ has property } P \}$$

This denotes the collection of exactly those objects x which satisfy property P. For example:

- $\{ b : b \text{ is a beach on Gower Peninsula} \}$
- $\{ p : p \text{ has climbed Mount Kailash} \}$
- $\{ n : n \text{ is a prime number} \}$
- $\{A\!: A \text{ is a set of people who share a common grandmother} \}$

Example Sets

$\emptyset = \{\,\}$	(empty set)
$\mathbb{B} = \{0, 1\}$	(binary digits, or bits)
$\mathbb{N} = \{0, 1, 2, 3, \ldots\}$	(natural numbers)
$\mathbb{Z} = \{\ldots, -2, -1, 0, 1, 2, \ldots\}$	(integers)
$\mathbb{Q} = \{ \frac{m}{n} : m, n \in \mathbb{Z}, n \neq 0 \}$	(rational numbers)
$\mathbb{R} = \{ x : x \text{ is a real number} \}$	(real numbers)

Note: $\emptyset$ and $\{\emptyset\}$ are very different sets!

Exercise: Write out the following sets explicitly, by listing their elements within curly braces.

1. $\{ x : x \text{ is an odd integer with } 0 < x < 8 \}$.
2. $\{ p : p \text{ is an even prime number} \}$.
3. $\{ x : x \text{ is a day of the week not containing the letter n} \}$.
4. $\{ d : d \text{ is a nephew of Donald Duck} \}$.

Membership of a Set

Membership of sets is denoted by $\in$, pronounced “is an *element* (or a *member*) of”, whilst *non-membership* is denoted by $\notin$.

Examples:

- $7 \in \{3, 7, 14\}$
- $8 \notin \{3, 7, 14\}$
- $\text{Felix} \in \{ \text{Joel}, \text{Felix}, \text{Oskar} \}$
- $\text{Amanda} \notin \{ \text{Joel}, \text{Felix}, \text{Oskar} \}$

Note: $x \notin A$ is the same as $\neg(x \in A)$.

Exercise: Which of the following propositions are true?

1. $2 \in \{1, 2, 3\}$.
2. $\{2\} \in \{1, 2, 3\}$.
3. $\{2\} \in \{\{1\}, \{2\}, \{3\}\}$.
4. $\emptyset \in \{\}$.
5. $\emptyset \in \{\emptyset\}$.

Set Equality

Two sets are ***equal*** if, and only if, they have the same elements.

That is to say, $A = B$ if for any x:

$$x \in A \quad \text{if, and only if,} \quad x \in B.$$

Examples:

$$\{3, 7, 14\} = \{7, 14, 3, 7, 3\}$$
$$\{\text{Joel, Felix, Oskar}\} \neq \{\text{Joel, Felix, Oskar, Amanda}\}$$

Exercise: Which of the following sets are equal?

- $A = \{1, \{1, 2\}\}$
- $B = \{1, \{2\}\}$
- $C = \{1, \{1\}\}$
- $D = \{\{1, 1\}, 1\}$
- $E = \{\{2, 1\}, 1\}$

Subsets

Set A is a ***subset*** of set B, if, and only if, each element of A is an element of B; in such case we write $A \subseteq B$.

We also say:

- B is a ***superset*** of A, written $B \supseteq A$;
- A is *included*, or *contained*, in B;
- B *includes*, or *contains*, A.

Further notation:

- $A \not\subseteq B$ denotes that A is *not* a subset of B, which means there is an element in A which is not in B.

  Note: $A \not\subseteq B$ is the same as $\neg(A \subseteq B)$.

- $A \subset B$ denotes that A is a ***proper subset*** of B, that is $A \subseteq B$ but $A \neq B$.

Example: $\emptyset \subset \mathbb{B} \subset \mathbb{N} \subset \mathbb{Z} \subset \mathbb{Q} \subset \mathbb{R}$.

Venn Diagrams

Venn diagrams are a graphical way of depicting sets.

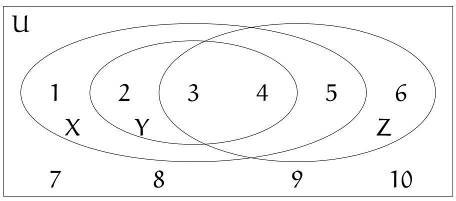
Consider the sets

$$X = \{1, 2, 3, 4, 5\} \qquad Y = \{2, 3, 4\} \qquad Z = \{3, 4, 5, 6\}$$

collecting objects from some universal set U, referred to as the ***universe of discourse***:

$$U = \{1, 2, \ldots, 9, 10\}$$

These sets can be depicted as follows:



Clearly:

- $Y \subseteq X$
- $X \not\subseteq Z$
- $Y \subset X$ since $1 \in X$ but $1 \notin Y$
- Z is is incomparable to both X and Y,
  since $X \not\subseteq Z$ and $Z \not\subseteq X$; and $Y \not\subseteq Z$ and $X \not\subseteq Y$.

Venn Diagrams

Using Venn diagrams we easily see that, for any sets A, B, C over a universe of discourse U:

- $\emptyset \subseteq A$ and $A \subseteq U$.
- If $A \subseteq B$ and $B \subseteq C$ then $A \subseteq C$.

Using Venn diagrams we also easily note the following properties of the subset relation $\subseteq$:

- It is ***reflexive***, meaning that: $A \subseteq A$.
- It is ***antisymmetric***, meaning that:

  if $A \subseteq B$ and $B \subseteq A$ then $A = B$.

- It is ***transitive***, meaning that:

  if $A \subseteq B$ and $B \subseteq C$ then $A \subseteq C$.

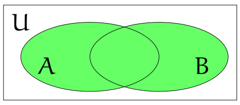
Moreover, $\emptyset$ is *smallest* set with respect to inclusion; that is, it is included in every other set: $\emptyset \subseteq A$ for every set A.

Exercise: Which of the following propositions are true?

1. $\{2\} \subseteq \{1, 2, 3\}$.
2. $\{1, 2, 3\} \subseteq \{\{1\}, \{2\}, \{3\}\}$.
3. $\{\{1, 2\}\} \subseteq \{\{1, 2, 3\}\}$.

Set Union

The **union** of two sets is the set consisting of all of the elements of the two sets taken together. Pictorially:



$A \cup B$

That is: $A \cup B = \{x : x \in A \text{ or } x \in B\}$

Thus: $x \in A \cup B \;\Leftrightarrow\; x \in A \vee x \in B$

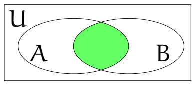
Example:

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

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

Set Intersection

The **intersection** of two sets is the set consisting of all of the elements that are in *both* of the two sets. Pictorially:



$A \cap B$

That is: $A \cap B = \{x : x \in A \text{ and } x \in B\}$

Thus: $x \in A \cap B \;\Leftrightarrow\; x \in A \wedge x \in B$

Example:

If $A = \{1, 2, 3, 4, 5\}$, $B = \{2, 4, 6, 8\}$, $C = \{3, 6, 9\}$

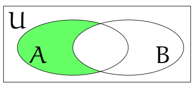
then
$$\begin{aligned} A \cap B &= \{2, 4\} \\ A \cap C &= \{3\} \\ B \cap C &= \{6\} \\ A \cap B \cap C &= \emptyset \end{aligned}$$

Inclusion-Exclusion Principle: For finite sets A and B:

$$|A \cup B| = |A| + |B| - |A \cap B|.$$

Set Difference

The **difference** of two sets is the set consisting of all of the elements of the first that are *not* in the second. Pictorially:



$A \setminus B$

That is: $A \setminus B = \{x : x \in A \text{ and } x \notin B\}$

Thus: $x \in A \setminus B \Leftrightarrow x \in A \wedge x \notin B$

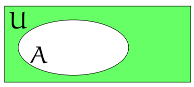
Example:

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

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

Set Complement

The **complement** of a set is the set consisting of all of the elements that are *not* in the set. Pictorially:



$\overline{A}$

Note: the complement operation is taken with respect to as underlying universe of discourse U.

That is: $\overline{A} = \{x : x \notin A\}$

Thus: $x \in \overline{A} \;\Leftrightarrow\; x \notin A$

Example: Assume $U = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$.

If $A = \{1, 2, 3, 4, 5\}$, $B = \{2, 4, 6, 8\}$, $C = \{3, 6, 9\}$

then $\overline{A} = \{6, 7, 8, 9\}$, $\overline{B} = \{1, 3, 5, 7, 9\}$, $\overline{C} = \{1, 2, 4, 5, 7, 8\}$

Set Operations

Exercise: Consider the following sets:

$$
\begin{aligned}
U &= \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\} \\
A &= \{1, 3, 5, 7, 9\} \\
B &= \{3, 4, 5\} \\
C &= \{5, 6, 7, 8, 9\}.
\end{aligned}
$$

Draw a Venn diagram depicting these sets.

Compute the following sets:

1. $A \cap B$
2. $(A \cap B) \cup C$
3. $A \cap (B \cup C)$
4. $(A \cup B) \setminus C$
5. $\overline{(A \cup B)} \cap C$

Powerset

The ***powerset*** $\mathcal{P}(A)$ of a set A is the set consisting of all subsets of A.

That is: $\mathcal{P}(A) = \{ X : X \subseteq A \}$

Thus: $x \in \mathcal{P}(A) \quad \Leftrightarrow \quad x \subseteq A$

In particular: $\emptyset \in \mathcal{P}(A)$ and $A \in \mathcal{P}(A)$

Examples:

1. $\mathcal{P}(\{0, 1\}) = \{ \emptyset, \{0\}, \{1\}, \{0, 1\} \}$
2. $\mathcal{P}(\{ \text{Joel, Felix, Oskar, Amanda} \})$
   $= \{ \emptyset,$
   $\{ \text{Joel} \}, \{ \text{Felix} \}, \{ \text{Oskar} \}, \{ \text{Amanda} \},$
   $\{ \text{Joel, Felix} \}, \{ \text{Joel, Oskar} \}, \{ \text{Joel, Amanda} \},$
   $\{ \text{Felix, Oskar} \}, \{ \text{Felix, Amanda} \}, \{ \text{Oskar, Amanda} \},$
   $\{ \text{Joel, Felix, Oskar} \}, \{ \text{Joel, Felix, Amanda} \},$
   $\{ \text{Joel, Oskar, Amanda} \}, \{ \text{Felix, Oskar, Amanda} \},$
   $\{ \text{Joel, Felix, Oskar, Amanda} \} \}$
3. $\mathcal{P}(\emptyset) = \{ \emptyset \}$ and $\mathcal{P}(\{a\}) = \{ \emptyset, \{a\} \}$
4. If $|A| = n$ then $|\mathcal{P}(A)| = 2^n$

Sets and Properties

We typically use set-building notation $\{ x : x \text{ has property } P \}$ to define a subset of a given set A, in which case we write:

$$B = \{ x \in A : x \text{ has property } P \}$$

instead of $B = \{ x : x \in A \text{ and } x \text{ has } P \}$

Example: Instead of defining set difference as

$$A \setminus B = \{ x : x \in A \text{ and } x \notin B \}$$

we can write

$$A \setminus B = \{ x \in A : x \notin B \}$$

Example: Given $a, b \in \mathbb{R}$, the following notations are frequently used for denoting intervals:

$$
\begin{aligned}
[a, b] &= \{ x \in \mathbb{R} : a \leq x \leq b \} \\
(a1, b1] &= \{ x \in \mathbb{R} : a < x \leq b \} \\
[a2, b2) &= \{ x \in \mathbb{R} : a \leq x < b \} \\
(a3, b3) &= \{ x \in \mathbb{R} : a < x < b \} \\
[m \ldots n] &= \{ k \in \mathbb{Z} : m \leq k \leq n \}
\end{aligned}
$$

In all cases: if left-hand value is greater than the right hand value, then the interval denotes the empty set $\emptyset$.

Russell's Paradox

Set-building notation is useful, but must be used with care.

Say a set A is *normal* if it is not an element of itself: $A \notin A$.

Consider the set R of all normal sets: $R = \{ A : A \notin A \}$.

Is R normal? That is, is the statement $R \in R$ true or false?

- If $R \in R$, then R must have the property which defines R, that is, $R \notin R$.
- If $R \notin R$, then R must *fail* to satisfy the property which defines R, that is $\neg(R \notin R)$; i.e., $R \in R$.

This anomaly is known as *Russell's Paradox*, after the philosopher Bertrand Russell.

If we use set-building notation only in restricted form:

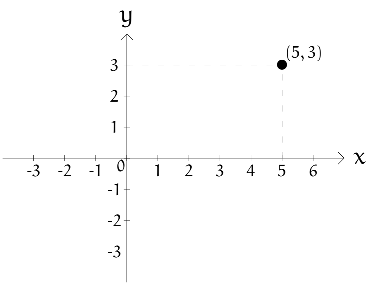
$$R = \{ x \in A : x \text{ has property } P \}$$

then we will never encounter such anomalies.

Ordered Pairs, Triples, n-tuples

An ***(ordered) pair*** is a comma-separated pair of objects (a, b) written within parantheses. The objects a and b are referred to as *coordinates*.

Example: Points in xy-plane are denoted by ordered pairs; e.g., the pair $(5, 3)$ refers to the point corresponding to 5 on the horizontal axis and 3 on the vertical axis.



Two pairs are equal if, and only if, their respective coordinates are equal:

$$(a, b) = (c, d) \quad \Leftrightarrow \quad a = c \;\wedge\; b = d$$

We can also have ***triples***, and more generally ***n-tuples*** consisting of an arbitrary number n of coordinates.

Cartesian Products

Given sets A and B, the set of all pairs (a, b) with $a \in A$ and $b \in B$ is denoted by $A \times B$:

$$A \times B \;=\; \{(a, b) : \; a \in A \text{ and } b \in B\}$$

Thus:

$$(a, b) \in A \times B \;\Leftrightarrow\; a \in A \;\wedge\; b \in B$$

Example: The Cartesian product $[1 \dots m] \times [1 \dots n]$ can model a finite grid, such as the points of an LCD screen or the squares of a chess board.

$$\begin{array}{cccc} (1,1) & (1,2) & \dots & (1,n) \\ (2,1) & (2,2) & \dots & (2,n) \\ \vdots & \vdots & \ddots & \vdots \\ (m,1) & (m,2) & \dots & (m,n) \end{array}$$

Note: The number of elements in product is the product of the number of elements of the individual sets.

$$|A \times B| \;=\; |A| \times |B|.$$

In particular:

$$A \times \emptyset \;=\; \emptyset \times A \;=\; \emptyset.$$

Cartsian Products

We can form the Cartesian product of any number of sets:

$$\begin{aligned} & A_1 \times A_2 \times \dots \times A_n \\ & \quad = \{(a_1, a_2, \dots, a_n) : \; a_1 \in A_1, a_2 \in A_2, \dots, a_n \in A_n\} \end{aligned}$$

We write A^n for $A \times A \times \dots \times A$

Example: $\mathbb{R}^3 = \mathbb{R} \times \mathbb{R} \times \mathbb{R}$ defines three-dimensional space.

Example Let S represent all students, C all courses, and G all possible grades. Then

$$S \times C \times G$$

represents all triples (s, c, g) where $s \in S$ is a student, $c \in C$ is a course, and $g \in G$ is a grade. A University student database would be a subset of $S \times C \times G$, recording the grades for all students registered in each course.

Example An address book would be represented by a subset of

Name $\times$ *Address* $\times$ *Phone*

where *Name*, *Address* and *Phone* are the sets of all possible Names, Addresses and Phone Numbers.

Modelling with Sets

Exercise: A Lewis Carroll puzzle assumes the following:

- All babies are illogical.
- Nobody is despised who can manage a crocodile
- Illogical persons are despised.

We are to deduce that no baby can manage a crocodile, using an appropriate Venn diagram.

Exercise: Use an appropriate Venn diagram to determine whether or not the following argument is valid.

- All oceans are full of water.
- No ponds are oceans.
- Therefore, no ponds are full of water.

Predicate Logic

Not all valid deductions can be formalised in propositional logic.

Example: How could you express the following logical deduction:

1. All horses are animals.
2. *Therefore,* all horse-heads are animal heads.

To obtain more expressivity, we extend propositional logic to include ***predicates***: properties which are true or false of elements of a universe of discourse.

We have actually been using predicates with set-building notation

$$\{ x : x \textit{ has property } P \}$$

The part "x *has property* P" is a predicate, often denoted $P(x)$. A predicate is an *indeterminate proposition*, which may be true or false of a particular element.

Predicates and Free Variables

Example:

$$\mathrm{Prime}(x) \quad = \quad \text{"}x\text{ is a prime number"}$$

is a predicate which says that x is a prime number. The universe of discourse is probably $\mathbb{N}$; but it could be anything (in which case $\mathrm{Prime}(x)$ would be false if $x \notin \mathbb{N}$).

The "x" in "$\mathrm{Prime}(x)$" is called a ***free variable*** as it stands for an unspecified value. If we instantiate x, we would get a proposition. For example

$\mathrm{Prime}(5)$ is true whereas $\mathrm{Prime}(6)$ is false

Example: Given the predicate $\mathrm{Female}(x) = $ "x is a female" where the universe of discourse is the Duck family:

$$\begin{aligned} \mathrm{Ducks} = \{ &\mathrm{Quackmore, Hortense, Scrooge,} \\ &\mathrm{Donald, Della, Huey, Louis, Dewey} \} \end{aligned}$$

Then

- $\mathrm{Female}(\mathrm{Hortense})$ and $\mathrm{Female}(\mathrm{Della})$ are true;
- $\mathrm{Female}(\mathrm{Quackmore})$, $\mathrm{Female}(\mathrm{Scrooge})$, $\mathrm{Female}(\mathrm{Donald})$, $\mathrm{Female}(\mathrm{Huey})$, $\mathrm{Female}(\mathrm{Louis})$ and $\mathrm{Female}(\mathrm{Dewey})$ are false.

Truth Sets

The ***truth set*** of a predicate $P(x)$ is the set of objects which satisfy it: $\{ x : P(x) \}$

Example: The truth set of $\mathrm{Prime}(x)$ is the set of all prime numbers $\{ 2, 3, 5, 7, 11, \ldots \}$.

Example: The truth set of the predicate $\mathrm{Female}(x)$ over the Duck family is $\{ \mathrm{Hortense}, \mathrm{Della} \}$.

Exercise: What are the truth sets of the following predicates?

1. $\mathrm{Even}(x) = $"$x$ is an even integer."
2. $\mathrm{EvenPrime}(x) = $"$x$ is an even prime number."

Predicates with two or more free variables

Predicates may range over more than one element.

For example, *equality* ($=$) and *set inclusion* ($\subseteq$) are binary predicates. For these, we typically use ***infix notation*** instead of ***prefix notation***; that is, $x{=}y$ instead of $=(x, y)$, and $A{\subseteq}B$ instead of $\subseteq(A, B)$.

For example: $5{=}5$ is true, $\emptyset{\subseteq}\{5\}$ is true, $\{\emptyset\}{=}\emptyset$ is false.

The truth set of a predicate which ranges over more than one element consists of tuples of values.

Example: Let $\mathrm{Divides}(x, y)$ be the two-place predicate over integers saying that x divides evenly into y. Then

- $\mathrm{Divides}(3, 15)$ is true, and $\mathrm{Divides}(4, 15)$ is false;
- the truth set of $\mathrm{Divides}(x, y)$ is

$$\{ (x, y) : x \text{ divides evenly into } y \}$$

(Note: The standard mathematical symbol for $\mathrm{Divides}$ is "|", used in infix notation, as in $3 \mid 15$ and $4 \nmid 15$.)

All and Some

We can form a proposition from a predicate by saying that it is true of *every* element, or that it is true of *some* (i.e., at least one) element.

Example: *"Every prime number greater than 2 is odd"*:

$$\mathrm{Odd}(3) \wedge \mathrm{Odd}(5) \wedge \mathrm{Odd}(7) \wedge \mathrm{Odd}(11) \wedge \cdots$$

where $\mathrm{Odd}(x)$ says "x is an odd number".

Example: *"Some prime numbers are square"*:

$$\mathrm{Square}(2) \vee \mathrm{Square}(3) \vee \mathrm{Square}(5) \vee \cdots$$

These are infinitely-long conjunctions / disjunctions, and cannot be expressed in propositional logic (without $\cdots$).

Predicate logic provides two forms of *quantification* which will allow to express these in a formula in a finite way: one which expresses that a property holds univerally (always), and another which expresses that a property holds existentially (sometimes).

Universal Quantification

The statement

$$\forall x\, P(x) \qquad (\text{pronounced } \textit{"for all } x, P(x)\textit{"})$$

is called ***universal quantification*** and is true if, and only if, $P(x)$ is true of *all* possible values of x.

Example: "*Everyone did the homework*" is expressed as:

$$\forall x\, H(x)$$

where $H(x)$ = "x did the homework", and the (assumed) universe of discourse is the set of students who were assigned the homework.

- "x" in the above formula is called a ***bound variable***; it is bound by the quantifier "$\forall x$".
- The statement has now a definite truth value.

Example: "*Every prime number greater than 2 is odd*" is expressed as:

$$\forall x \left(\mathrm{Prime}(x) \wedge x > 2 \Rightarrow \mathrm{Odd}(x) \right)$$

Note: The (outer) parentheses are needed, as universal quantification binds more strongly than all propositional connectives.

Universal Quantification

Example: *"Every dog that has stayed in the kennel will have to go into quarantine"* can be expressed as

$$\forall x \left(K(x) \Rightarrow Q(x) \right)$$

where $K(x)$ = "x has stayed in the kennel" and $Q(x)$ = "x has to go into quarantine". The (assumed) universe of discourse is the set of dogs.

Breaking this down:

For all x: if x has stayed in the kennel,
then x has to go in to quarantine.

Example: *"Nobody likes a sore loser"* can be expressed as:

$$\forall x \left(S(x) \Rightarrow \forall y \neg L(y, x) \right)$$

where $S(x)$ = "x is a sore loser" and $L(y, x)$ = "y likes x". The (assumed) universe of discourse is set of all people.

Breaking this down:

For all x: if x has is a sore loser,
then for all y: y doesn't like y.

Universal Quantification

Exercise: Using the predicates

- $B(x)$ = "x is a bee"
- $F(x)$ = "x is a flower"
- $L(x, y)$ = "x likes y"

write each of the following statements in predicate logic.

1. Bees like all flowers.

   (For all x: if x is a bee,
   then for all y: if y is a flower,
   then x likes y.)

2. Bees only like flowers.

   (For all x: if x is a bee,
   then for all y: if x likes y,
   then y is a flower.)

3. Only bees like flowers.

   (For all x and for all y: if y is a flower
   and x likes y
   then x is a bee.)

Existential Quantification

The statement

$\exists x\, P(x)$ (pronounced *"there exists x such that $P(x)$"*)

is called ***existential quantification*** and is true if, and only if, $P(x)$ is true of *some* (at least one) value of x.

Example: *"Some prime numbers are square"* can be expressed as:

$$\exists x\Big(\mathrm{Prime}(x) \land \mathrm{Square}(x)\Big)$$

- "x" is again called a ***bound variable***,

  it is bound by the quantifier "$\exists x$".

- The statement has a definite truth value (it is false).

Example: *"Someone didn't do the homework"* is expressed as:

$$\exists x\, \neg H(x)$$

where $H(x)$ = "x did the homework", and the (assumed) universe of discourse is the set of students who were assigned to do the homework.

Note: Existential quantification also binds more strongly than all propositional connectives.

Univesal Quantification

Example: *"If some dog that has stayed in the kennel has been in contact with a dog with rabies, then every dog that has stayed in the kennel will have to go into quarantine"* can be expressed as

$$\exists x\Big(K(x) \land \exists y\,(C(x,y) \land R(y))\Big) \Rightarrow \forall x\Big(K(x) \Rightarrow Q(x)\Big)$$

where
$$\begin{aligned} K(x) &= \text{"}x \text{ has stayed in the kennel"} \\ R(x) &= \text{"}x \text{ has rabies"} \\ C(x,y) &= \text{"}x \text{ and } y \text{ have been in contact"} \\ Q(x) &= \text{"}x \text{ has to go into quarantine"} \end{aligned}$$

The (assumed) universe of discourse is the set of dogs.

Breaking this down:

If there is an x that has stayed in the kennel,
and there is a y that
has been in contact with x and has rabies,
then for all x: if x has stayed in the kennel,
then x has to go into quarantine.

Universal Quantification

Exercise: Let $M(x,y)$ say *"x is the mother of y"*, where the universe of discourse is the set of human beings.

Express the following predicates

1. Every human being has a mother.

   (For each x there is a y such that
   y is the mother of x.)

2. Every human being has exactly one mother.

   (For each x there is a y such that
   y is the mother of x,
   and for every z: if z is the mother of x,
   then $z = x$.)

Bounded Quantification

The following are useful restricted forms of quantification which we use for convenience:

- $\forall x{\in}A\; P(x)$ pronounced *"for all x in A, $P(x)$"*
  logically equivalent to: $\forall x\,\big(x{\in}A \Rightarrow P(x)\big)$

- $\exists x{\in}A\; P(x)$ pronounced *"there exists some x in A such that $P(x)$"*
  logically equivalent to: $\exists x\,\big(x{\in}A \land P(x)\big)$

- $\exists! x\, P(x)$ pronounced *"there exists a unique x such that $P(x)$"*
  logically equivalent to: $\exists x\,\big(P(x) \land \neg\exists y (P(y) \land x{\neq}y)\big)$

Exercise: Let $T(s,c)$ denote *"student s takes course c"*. Express the following statements in predicate logic.

1. Alice and Bob take exactly one course together.
2. Alice and Bob take exactly two courses together.

Rules for Quantification

- If it is not the case that $P(x)$ is true for all x, then there must be some x for which $P(x)$ is false:

$$\neg\forall x\, P(x) \;\Leftrightarrow\; \exists x\, \neg P(x)$$

- If it is not the case that $P(x)$ is true for some x, then $P(x)$ must be false for all x:

$$\neg\exists x\, P(x) \;\Leftrightarrow\; \forall x\, \neg P(x)$$

Exercise: For each of the following statements, identify which option correctly expresses its negation. Translate each statement into predicate logic to confirm your choice.

1. Some people like mathematics.
   (a) Some people dislike mathematics.
   (b) Everybody dislikes mathematics.
   (c) Everybody likes mathematics.
2. All cats have fur and a tail.
   (a) No cats has fur and a tail.
   (b) Some cats are bald and tailless.
   (c) Some cats are bald or tailless.

Modelling in Predicate Logic

Example: A Lewis Carroll puzzle. Deduce that no baby can manage a crocodile from the following assumptions:

- All babies are illogical.
- Nobody is despised who can manage a crocodile
- Illogical persons are despised.

Define the following predicates:

$B(x)$ = "x is a baby" $D(x)$ = "x is despised"
$I(x)$ = "x is illogical" $M(x)$ = "x can manage a crocodile"

The three premises above translate into the following:

1. $\forall x \left(B(x) \Rightarrow I(x)\right)$
2. $\forall x \left(M(x) \Rightarrow \neg D(x)\right)$ or $\forall x \left(D(x) \Rightarrow \neg M(x)\right)$
3. $\forall x \left(I(x) \Rightarrow D(x)\right)$

and the conclusion translates to $\forall x \left(B(x) \Rightarrow \neg M(x)\right)$.

For any x for which $B(x)$ is true, we have $I(x)$ true by 1., then $D(x)$ true by 3., and finally $\neg M(x)$ true by 3.

Hence, the conclusion indeed follows from the premises.

Modelling in Predicate Logic

Example: Recall our earlier deduction:

1. All horses are animals.
2. *Therefore,* all horse-heads are animal heads.

Letting

- $\mathrm{Horse}(x)$ = "x is a horse"
- $\mathrm{Animal}(x)$ = "x is an animal" and
- $\mathrm{Head}(x, y)$ = "x is the head of y"

the statements above translate to:

1. $\forall x \left(\mathrm{Horse}(x) \Rightarrow \mathrm{Animal}(x)\right)$
2. $\forall x \left(\exists y (\mathrm{Horse}(y) \wedge \mathrm{Head}(x, y)) \Rightarrow \exists y (\mathrm{Animal}(y) \wedge \mathrm{Head}(x, y))\right)$

Modelling in Predicate Logic

Exercise: Formalise the following two arguments in predicate logic, discuss any ambiguities, and decide whether they are valid.

1. Everybody loves somebody.
   Therefore, somebody is loved by everybody.

2. Somebody loves everybody.
   Therefore, everybody is loved by somebody.

Functions

A ***function*** f from a set A to a set B, denoted

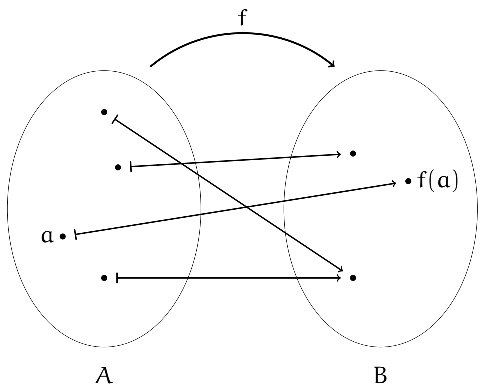
$$f : A \to B$$

associates a *unique* value of B to *each* value of A.

$f(a)$ refers to the element of B assigned to $a \in A$ by f.

Thus, f maps each element $a \in A$ to an element $b = f(a) \in B$, which we also denote by $f\colon a \mapsto b$.

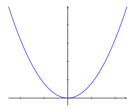
Pictorially:



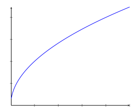
Functions

Functions are familiar in mathematics. For example:

- The squaring function $sq : \mathbb{R} \to \mathbb{R}$ takes a real value and squares it, and is typically written as $sq(x) = x^2$.



- The square root function $sqrt : \mathbb{R}^{\geq 0} \to \mathbb{R}^{\geq 0}$ takes a non-negative real number and returns its square root, and is typically written as $sqrt(x) = \sqrt{x}$.



- The function *floor* : $\mathbb{R} \to \mathbb{Z}$ takes a real number and return the largest integer which does not exceed the given real value; this is typically written $floor(x) = \lfloor x \rfloor$. Thus for example $floor(3.8)=3$ and $floor(-3.8)=-4$.

Functions

Example: Grades (in the form of integer percentages) of the students in a class:

Archer	75	Evans	78	Parks	64
Byron	92	Farmer	46	Smith	59
Cole	64	Greene	68	Taylor	100
Davies	88	Lewis	54	Wells	78

Each person in the set

$$\begin{aligned} \mathrm{Class} = \{&\mathrm{Archer, Byron, Cole, Davies, Evans, Farmer} \\ &\mathrm{Greene, Lewis, Parks, Smith, Taylor, Wells}\} \end{aligned}$$

is assigned a value from set

$$\mathrm{Marks} = \{0, 1, 2, 3, \ldots, 100\}$$

This describes a function

$$\mathrm{score}\colon \mathrm{Class} \to \mathrm{Marks}$$

For example, $\mathrm{score(Greene)} = 68$;

- the function score maps $\mathrm{Greene} \in \mathrm{Class}$ to $68 \in \mathrm{Marks}$,
- that is, $\mathrm{score}\colon \mathrm{Greene} \mapsto 68$.

Functions

$f\colon A \to B$ assigns exactly one value $f(a) \in B$ to each $a \in A$.

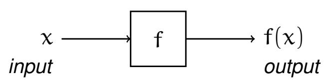
- For example, in the score function above, each student has a single well-defined score.

However, it may assign the same value of B to different values of A.

- For example, in the score function above,

$$\mathrm{score(Cole)} = \mathrm{score(Parks)} = 64$$

Thus, f is like machine into which you feed a value x and it responds by outputting a value $f(x)$:



Exercise: Which of the following are valid functions from the set of humans to the set of humans?

1. *Mother* (x), representing the mother of x.
2. *Child* (x), representing the child of x.
3. *FirstChild* (x), representing the first-born child of x.

Domain, Codomain and Range

If $f\colon A \to B$, then

- A is called the ***domain*** of f.

  $$\mathrm{domain}(\mathrm{score}) = \mathrm{Class}$$

- B is called the ***codomain*** of f.

  $$\mathrm{codomain}(\mathrm{score}) = \mathrm{Marks}$$

- The ***range*** of f is the subset of the codomain consisting of the values that are produced by f:

  $$\mathrm{range}(f) = \{\, f(a) : a \in A \,\}$$

  $$\mathrm{range}(\mathrm{score}) = \{\, 46, 54, 59, 64, 68, 75, 78, 88, 92, 100 \,\}$$

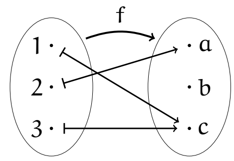
Given $S \subseteq A$, the ***image*** of S under f, denoted $f(S)$, is given as $f(S) = \{\, f(a) : a \in S \,\}$. Thus, $\mathrm{range}(f) = f(A)$.

Given $T \subseteq B$, the ***preimage*** of T under f, denoted $f^{-1}(T)$, is given as $f^{-1}(T) = \{\, a \in A : f(a) \in T \,\}$. Thus $f^{-1}(B) = A$.

Note then that $f : \mathcal{P}(A) \to \mathcal{P}(B)$ and $f^{-1} : \mathcal{P}(B) \to \mathcal{P}(A)$ can be viewed as functions on the powersets of A and B.

Functions

Example: Consider the function $f\colon \{1,2,3\} \to \{a,b,c\}$ defined by $f(1) = c$, $f(2) = a$, $f(3) = c$.



- The domain of f is $\{1,2,3\}$.
- The codomain of f is $\{a,b,c\}$.
- The range of f is $\{a,c\}$
- $f\big(\{1,2\}\big) = \{a,c\}$ and $f\big(\{1,3\}\big) = \{c\}$.
- $f^{-1}\big(\{b,c\}\big) = \{1,3\}$ and $f^{-1}\big(\{c\}\big) = \{1,3\}$.

Example: A first-class mark is a grade of 70 or higher. The set of students who have scored a first-class mark is thus

$$\mathrm{score}^{-1}\big(\{\, n \in \mathbb{N} \,:\, 70 \le n \le 100 \,\}\big).$$

Example Standard Functions

Example: Common functions for an arbitrary set A.

1. The *identity function* $\mathrm{id}_A\colon A \to A$ maps each element in A to itself:

   $$\mathrm{id}_A(x) = x$$

2. The *cardinality function* $|\cdot|\colon \mathcal{P}_{\mathrm{fin}}(A) \to \mathbb{N}$ maps each finite subset of A to the number of elements in that subset:

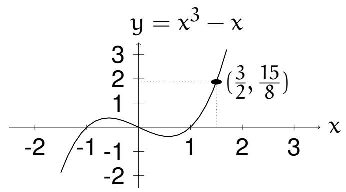
   $|X|$ = number of elements in X

3. Given a subset $S \subseteq A$ of A, its *characteristic function* $\chi_S\colon A \to \mathbb{B}$ indicates whether or not an object is an element of S:

   $$\chi_S(x) = \begin{cases} 1, & \text{if } x \in S \\ 0, & \text{if } x \notin S \end{cases}$$

Graph of function

Functions in mathematics are plotted as a graph:



The example above shows that we can represent a function $f\colon A \to B$ by its *graph*:

$$\mathrm{graph}(f) = \big\{\, (a,b) \in A \times B : b = f(a) \,\big\}$$

So for $f(x) = x^3 - x$: $\mathrm{graph}(f) = \big\{\, (x, x^3 - x) : x \in \mathbb{R} \,\big\}$

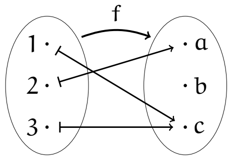
and $\mathrm{graph}(\mathrm{score}) = \big\{$ (Archer, 75), (Byron, 92), (Cole, 64), (Davies, 88), (Evans, 78), (Farmer, 46), (Greene, 68), (Lewis, 54), (Parks, 64), (Smith, 59), (Taylor, 100), (Wells, 78) $\big\}$

Two functions f and g are equal if, and only if, they share the same domain and codomain, and $\mathrm{graph}(f) = \mathrm{graph}(g)$.

Finctions and their Graphs

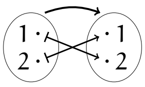
Exercise: What is the graph of the following function:

f

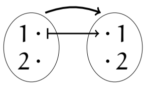


Exercise: Let $X = \{1, 2\}$. Which of the following descriptions define a function from X to X?

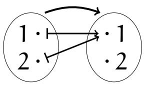
f_1



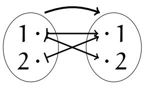
f_2



f_3



f_4



For those which are functions, determine their graphs.

Exercise: Find all other valid functions from X to X, and compute their graphs.

One-To-One Functions

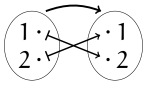
A function $f \colon A \to B$ associates a single $b \in B$ to each $a \in A$, but the same $b \in B$ may be associated to several values of A. For example, for $f(x) = x^3 - x$ we have $f(-1)=0$ and $f(0)=0$ and $f(1) = 0$.

If f does *not* assign the same value to different inputs, it is called ***one-to-one*** (1-1), or ***injective***. That is,

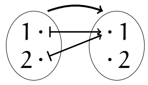
$$\forall\, a, a' \in A \; \big(f(a) = f(a') \;\Rightarrow\; a = a'\big).$$

Example: Consider functions f_1 and f_3 from before:

f_1



f_3



f_1 is injective; but f_3 is not injective, as $f_3(1) = f_3(2) = 1$.

Exercise: Which of the following functions are one-to-one? Explain why the others aren't.

1. The function $\mathrm{score} \colon \mathrm{Class} \to \mathrm{Marks}$.
2. The function $f \colon \mathbb{N} \to \mathbb{N}$ defined by $f(x) = x^2$.
3. The function $f \colon \mathbb{R} \to \mathbb{R}$ defined by $f(x) = x^2$.
4. The function $f \colon \mathbb{R} \to \mathbb{R}$ defined by $f(x) = x^3$.

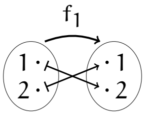
Onto Functions

A function $f \colon A \to B$ is ***onto***, or ***surjective***, if, and only if, every value in its codomain is produced by the function on some input; that is, its range equals its codomain:

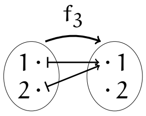
$$\forall b \in B \; \exists a \in A \; \big(f(a) = b\big).$$

Example: Consider functions f_1 and f_3 from before:

f_1



f_3



f_1 is onto; but f_3 is not onto, as 2 is not in its range.

Exercise: Which of the following functions are onto? Explain why the others aren't.

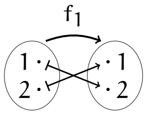
1. The function $\mathrm{score} \colon \mathrm{Class} \to \mathrm{Marks}$.
2. The function $f \colon \mathbb{N} \to \mathbb{N}$ defined by $f(x) = x^2$.
3. The function $f \colon \mathbb{R} \to \mathbb{R}$ defined by $f(x) = x^2$.
4. The function $f \colon \mathbb{R} \to \mathbb{R}$ defined by $f(x) = x^3$.

Bijective Functions

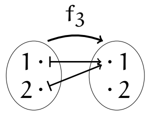
A function $f \colon A \to B$ is ***bijective***, or a ***bijection***, if, and only if, it is both one-to-one and onto.

Example: Consider functions f_1 and f_3 from before:

f_1



f_3



f_1 is bijective; but f_3 is not bijective, as it is neither one-to-one nor onto.

Exercise: Which of the following functions are bijections? Explain why the others aren't.

1. The function $\mathrm{score} \colon \mathrm{Class} \to \mathrm{Marks}$.
2. The function $f \colon \mathbb{N} \to \mathbb{N}$ defined by $f(x) = x^2$.
3. The function $f \colon \mathbb{R} \to \mathbb{R}$ defined by $f(x) = x^2$.
4. The function $f \colon \mathbb{R} \to \mathbb{R}$ defined by $f(x) = x^3$.

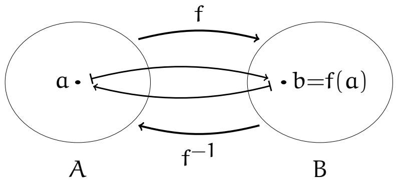
Inverting a Bijection

If $f: A \to B$ is a bijection, then

- f is onto: every $b \in B$ is image of *some* $a \in A$; and
- f is one-to-one: every $b \in B$ is image of a *unique* $a \in A$.

The ***inverse function*** $f^{-1}: B \to A$ is the function that assigns to each $b \in B$ the unique $a \in A$ such that $f(a)=b$.

$$f^{-1}(b) = a \ \text{ if, and only if, } \ f(a)=b$$



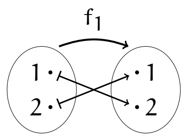
$f^{-1}: B \to A$ is also a bijection, and $(f^{-1})^{-1} = f$.

Example: Consider the bijection f_1 from above:

We compute

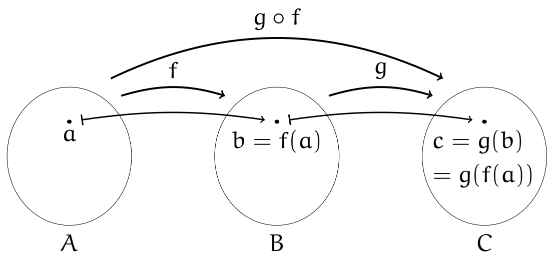
$$f_1^{-1}(1) = 2 \quad \text{and} \quad f_1^{-1}(2) = 1$$

(Thus, in fact, $f_1^{-1} = f_1$.)



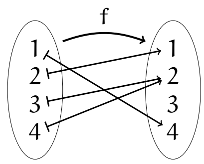
Composing Functions

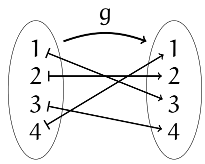
If we have two functions $f: A \to B$ and $g: B \to C$, then we can ***compose*** these two functions – i.e., apply f followed by g – which gives us a new function $g \circ f : A \to C$:



Given $f: A \to B$ and $g: B \to C$, the ***composition*** of g and f is the function $g \circ f: A \to C$ defined by $(g \circ f)(x) = g(f(x))$.

Exercise: Let $A = \{1, 2, 3, 4\}$ and consider $f, g: A \to A$:





Find $f \circ g$, $g \circ f$, $f \circ f$ and $g \circ g$.

Composing Functions

If $f: A \to A$ then we can form $f \circ f$, which we denote by f^2.

More generally, f^n applies the function f n times in a row:

$$\begin{aligned}
f^0 &= \mathrm{id}_A \\
f^1 &= f \\
f^2 &= f \circ f \\
f^3 &= f \circ f \circ f \\
f^n &= \underbrace{f \circ \cdots \circ f}_{n \text{ times}}
\end{aligned}$$

Note:

- If $f : A \to B$ and $g : B \to C$ are both one-to-one,

  then so is $g \circ f : A \to C$.

- If $f : A \to B$ and $g : B \to C$ are both onto,

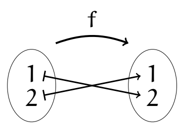
  then so is $g \circ f : A \to C$.

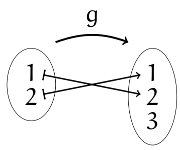
- If $f : A \to B$ and $g : B \to C$ are both bijective,

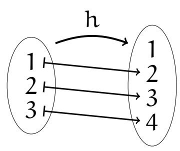
  then so is $g \circ f : A \to C$.

Composing Functions

Exercise: Consider the following functions:







Which of the following compositions are defined? For each of these, compute its domain, codomain and range.

1. $g \circ f$	4. $g \circ g$	7. $h \circ f$
2. $f \circ g$	5. $h \circ g$	8. $f \circ h$
3. $f \circ f$	6. $g \circ h$	9. $h \circ g \circ f$

Relations

A function $f\colon A \to B$ maps elements from set A to elements in another set B.

We will now study more general *relationships* between elements than simple mappings.

Examples:

- "parenthood" amongst the set of people ("x is parent of y")
- "divisibility" amongst the set of integers ("x divides evenly into y")
- relationships between different sets: "enrolment" relating the sets of students and courses ("student s takes course c")
- relationships between more than two sets: "grade" relating students, courses and grades ("students s got a grade of g in course c")

Relations from Predicates

Example: Consider the predicate

$\mathrm{Mark}(x, y, z) =$ "student x, in course y, scored grade z"

Let S, C and G be the sets of students, courses, and grades, respectively. The truth set of S is a subset Grades of the Cartesian product $S \times C \times G$ given as

$$\mathrm{Grades} = \{\, (s, c, g) : S(s, c, g) \,\}$$

An n***-ary relation*** R is a subset of n-tuples.

If $R \subseteq A_1 \times A_2 \times \ldots \times A_n$, we say that

R *is a relation over* $A_1 \times A_2 \times \ldots \times A_n$.

E.g., Grades is a ternary (that is, 3-ary) relation over $S \times C \times G$.

The IMDb Database

A small part of the Internet Movie Database (IMDb) representing James Bond Films is presented on the next slide.

The database can be viewed as a 5-ary relation:

$$\mathrm{IMDb007} \subseteq \mathrm{Titles} \times \mathbb{N} \times \mathrm{Names} \times \mathcal{P}_{\mathrm{fin}}(\mathrm{Names}) \times \mathrm{Names}.$$

where $\mathrm{Titles} =$ set of film titles in the database
$\mathrm{Names} =$ set of names of people in the database

An element in the database associates a title with a year of release, the actor playing James Bond, its screenwriters, and its director.

For example

$r10 = \big($ The Man With the Golden Gun,
1974,
Roger Moore,
$\{$Richard Maibaum, Tom Mankiewicz$\}$,
Guy Hamilton $\big)$.

IMDB Database

IMDb007	Title	Year	Starring Role	Screenwriters	Director
r01	Dr. No	1962	Sean Connery	Ian Fleming, Richard Maibaum	Terence Young
r02	From Russia with Love	1963	Sean Connery	Richard Maibaum	Terence Young
r03	Goldfinger	1964	Sean Connery	Richard Maibaum, Paul Dehn	Guy Hamilton
r04	Thunderball	1965	Sean Connery	Kevin McClory, Jack Whittingham	Terence Young
r05	You Only Live Twice	1967	Sean Connery	Roald Dahl, Harold Jack Bloom	Lewis Gilbert
r06	Casino Royale	1967	David Niven	Wolf Mankowitz, John Law, Michael Sayers	Val Guest
r07	On Her Majesty's Secret Service	1969	George Lazenby	Simon Raven, Richard Maibaum	Peter R. Hunt
r08	Diamonds Are Forever	1971	Sean Connery	Richard Maibaum, Tom Mankiewicz	Guy Hamilton
r09	Live and Let Die	1973	Roger Moore	Tom Mankiewicz	Guy Hamilton
r10	The Man with the Golden Gun	1974	Roger Moore	Richard Maibaum, Tom Mankiewicz	Guy Hamilton
r11	The Spy Who Loved Me	1977	Roger Moore	Christopher Wood, Richard Maibaum	Lewis Gilbert

Binary Relations

Binary (that is, 2-ary) relations are most common, often written in infix notation; i.e., as $a\,R\,b$ instead of $(a, b) \in R$.

Binary relations are common for expressing:

- order ("element a comes before element b)
- equivalence ("element a is the same as element b)
- functions ("input a results in output b)

A relation $R \subseteq A \times B$ is said to be a relation *from* A *to* B; the set A is the ***source*** of R, and the set B is its ***sink***.

If $R \subseteq A \times A$ – that is, the source is the same as the sink – then R is said to be a relation *on* A.

Examples:

- The *less-than-or-equal* relation $\leq \subseteq \mathbb{N} \times \mathbb{N}$:

$$\begin{aligned} \leq \; &= \{(x, y) : x \leq y\} \\ &= \{(0,0),(0,1),(1,1),(0,2),(1,2),(2,2),\ldots\}. \end{aligned}$$

- A relation $R \subseteq B \times \mathbb{N}$ from the set B of all living British people to the set $\mathbb{N}$ of natural numbers:

$$R = \{(b, n) : n \text{ is a bank account number of } b\}.$$

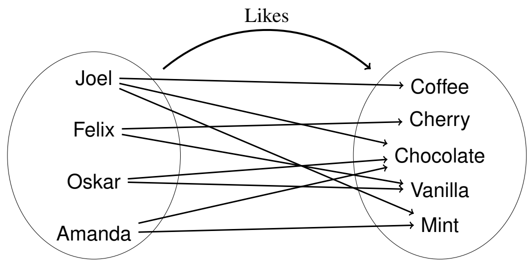
Binary Relations

Example: Joel likes mint and coffee ice cream; Felix likes vanilla and cherry ice cream; Oskar likes vanilla and chocolate ice cream; and Amanda likes chocolate and mint ice cream. These properties define a binary relation

$$\text{Likes} \subseteq \text{Children} \times \text{Flavours} \quad \textit{where}$$
$$\text{Children} = \{\text{Joel, Felix, Oskar, Amanda}\} \quad \textit{and}$$
$$\text{Flavours} = \{\text{Vanilla, Chocolate, Coffee, Cherry, Mint}\}$$

$\text{Likes} = \{$ (Joel, Mint), (Joel, Coffee), (Joel,Chocolate)
(Felix, Vanilla), (Felix, Cherry),
(Oskar, Vanilla), (Oskar, Chocolate),
(Amanda, Chocolate), (Amanda, Mint) $\}$.

We can visualise such a relation as follows:



Visualising Relations

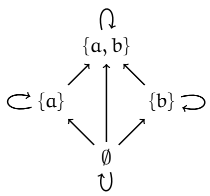
Binary relations from a set A to a set B can be visualised as above by drawing arrows from elements of the set A to related elements in the set B.

For homogeneous relations on a set A, we need only draw the set A once and add the relation arrows. $R \subseteq A \times A$, draw A as a set once, and an arrow from a to

Example: The subset relation $\subseteq$ on the powerset of $\{a, b\}$:

$$\mathcal{P}\big(\{a,b\}\big) \;=\; \{\emptyset, \{a\}, \{b\}, \{a,b\}\}$$

is pictured as follows:



Exercise: Let $A = \{$cat, dog, bird, rat$\}$, and let

$$R_A = \{(x, y) : x \text{ and } y \text{ have a letter in common}\}$$

Draw R_A.

Reflexive Relations

A relation R on a set A is ***reflexive*** if, and only if, *every element* of A is related to itself by R. That is,

$$\forall x \in A\ (x\,R\,x).$$

R is ***irreflexive*** if, and only if, *no* element of A is related to itself. That is,

$$\forall x \in A\ \neg(x\,R\,x).$$

Examples:

- The *less-than-or-equal-to* relation $\leq$ is reflexive.
- the *less-than* relation $<$ is irreflexive.

Note: Irreflexive is not the same as non-reflexive: there are relations which are neither reflexive nor irreflexive. However, no relation can be both reflexive and irreflexive (except for the empty relation over the empty set).

Exercise: Let $A = \{$cat, dog, bird, rat$\}$, and let

$$R_A = \{(x, y) : x \text{ and } y \text{ have a letter in common}\}$$

Is the relation R_A reflexive? Is it irreflexive?

Symmetric Relations

A relation R on a set A is ***symmetric*** if, and only if, y is related to x whenever x is related to y. That is,

$$\forall x, y \in A \left((x\, R\, y) \Rightarrow (y\, R\, x) \right).$$

R is ***antisymmetric*** if, and only if, y is *never* related to x if x is related to y, *except* possibly for when $x{=}y$. That is,

$$\forall x, y \in A \left((x\, R\, y) \wedge (y\, R\, x) \Rightarrow x{=}y \right).$$

Examples:

- The relations $<$ and $\leq$ are both antisymmetric, while
- the relation $=$ is symmetric (as well as antisymmetric).

Note: antisymmetric is not the same as non-symmetric: the relation $=$ is both symmetric and antisymmetric.

Exercise: Let $A = \{$cat, dog, bird, rat$\}$, and let

$$R_A = \{ (x, y) : x \text{ and } y \text{ have a letter in common} \}$$

Is the relation R_A symmetric? Is it antisymmetric?

Transitive Relations

A relation R on a set A is ***transitive*** if, and only if, x is related to z whenever x is related to some y which is related to z. That is,

$$\forall x, y, z \in A \left((x\, R\, y) \wedge (y\, R\, z) \Rightarrow (x\, R\, z) \right).$$

Examples:

- The relations $<$, $\leq$ and $=$ are all transitive.
- The sibling relationship over people is not reflexive, but irreflexive. It is symmetric but not antisymmetric. Finally, it is not transitive: Assume Joel and Felix are brothers, then Joel is a sibling of Felix and Felix is sibling of Joel, so if it were transitive Joel would have to be sibling of himself, which he is not.

Exercise:

- Let $A = \{$cat, dog, bird, rat$\}$, and let

  $$R_A = \{ (x, y) : x \text{ and } y \text{ have a letter in common} \}$$

  Is the relation R_A transitive?
- Is the *is-ancestor-of* relation over people reflexive? Is it irreflexive? Symmetric? Antisymmetric? Transitive?

Orderings Relations

A relation R on a set A is a ***partial order*** if, and only if, R is reflexive, antisymmetric and transitive.

R is a ***total order*** if, and only if, R is a partial order in which any two elements are related in one way or the other:

$$\forall x, y \in A \left((x\, R\, y) \vee (y\, R\, x) \right).$$

Example:

- $=$ on $\mathbb{Z}$ is a partial order but not a total order.
- $\leq$ on $\mathbb{Z}$ is a total order.
- $<$ on $\mathbb{Z}$ is not a partial order as it is not reflexive.
- $\subseteq$ on sets is a partial order but not a total order.

Exercise: Which of the following binary relations on $\mathbb{N}$ are partial orders? Which are total orders?

1. The identity relation $I = \{ (n, n) : n \in \mathbb{N} \}$.
2. The universal relation $U = \{ (m, n) : m, n \in \mathbb{N} \}$.
3. The parity relation $P = \{ (m, n) : m = n (\mathrm{mod} 2) \}$.

Equivalence Relations

A binary relation R on a set is an ***equivalence relation*** if, and only if, R is reflexive, symmetric and transitive.

Example:

- $=$ on $\mathbb{Z}$ is an equivalence relation.
- $\leq$ on $\mathbb{Z}$ is not an equivalence relation as it is not symmetric.
- $<$ on $\mathbb{Z}$ is not an equivalence relation as it is not symmetric and not reflexive.
- $\subseteq$ on sets is not an equivalence relation, as it is not symmetric.
- the *same-month-of-birth* relation on people – in which two people are related if, and only if, their birthdays are in the same month – is an equivalence relation.

Exercise: Which of the following binary relations on $\mathbb{N}$ are equivalences?

1. The identity relation $I = \{ (n, n) : n \in \mathbb{N} \}$.
2. The universal relation $U = \{ (m, n) : m, n \in \mathbb{N} \}$.
3. The parity relation $P = \{ (m, n) : m = n\, (\mathrm{mod}\, 2) \}$.

Equivalence Classes and Partitions

Consider the equivalence relation *same-month-of-birth*:

$\{ (x, y) : x$ and y have birthdays in the same month $\}$.

This equivalence relation partitions a set A of people into 12 groups depending on their birthday. For example, one group might consist of all those who have their birthday in November. There may be fewer than 12 groups, if there are months in which no one in A was born.

A ***partition*** of a set A is a collection

$$\{A_i : i \in I\}$$

of disjoint non-empty subsets of A, which together contain all of A. That is:

1. $A_i \cap A_j = \emptyset$ whenever $i \neq j$; and
2. $\forall x\ (x \in A \Leftrightarrow \exists i \in I\ x \in A_i)$.

Each A_i is called a ***block*** of the partition.

Example: over the set $A = \{1, 2, 3, 4\}$,

- $\{\{1, 2\}, \{3, 4\}\}$ and $\{\{1, 3, 4\}, \{2\}\}$ are partitions of A.
- $\{\{1\}, \{3, 4\}\}$ and $\{\{1, 2\}, \{2, 3, 4\}\}$ are not partitions of A.

Partition Refinement

We say that one One partition is a ***refinement*** of a second partition if, and only if, every block of the first is a subset of some block of the second.

Example: We can refine the *same-month-of-birth* relation into a *same-month-of-birth-and-same-sex* relation by also splitting the people in A also according to sex. Thus for example, one group might consist of all females whose birthday is in November.

Exercise: Over the set $A = \{a, b, c, d\}$, which of the following form a partition of A?

1. $\{ \{a\}, \{b, c\}, \{d\} \}$
2. $\{ \{a, c\}, \{b, d\} \}$
3. $\{ \{a, c\}, \{b\} \}$
4. $\{ \{a\}, \{b\}, \{c\}, \{d\} \}$
5. $\{ \{a, c\}, \{b\}, \{c, d\} \}$
6. $\{ \{a, b, c, d\} \}$

Exercise: How many partitions are there on the set $\{1, 2, 3\}$? Which is the finest (most refined) partition? Which is the coarsest (i.e., least fine) partition?

Equivalences from Partitions

A partition on a set A defines an equivalence relation R in the following way: two elements of A are in relation R if, and only if, they appear in the same block of the partition.

Exercise: Consider the partition $\{ \{1, 2\}, \{3, 4\} \}$ of $\{1, 2, 3, 4\}$. Describe the corresponding equivalence relation.

Conversely, as we've seen, any equivalence relation on A partitions A into disjoint non-empty subsets (blocks), called the ***equivalent classes*** of the equivalence.

Given an equivalence relation R on A, the *equivalence class* of $a \in A$ with respect to R, denoted $[a]_R$, is the set of elements of A related to a by R:

$$[a]_R = \{ x \in A : a\, R\, x \}.$$

The collection $\{ [a]_R : a \in A \}$ of these equivalence classes is a partition of A.

Exercise: Consider the following equivalence relation on $\{a, b, c, d\}$:

$$\{ (a, a),\ (b, b),\ (c, c),\ (d, d),\ (b, c),\ (c, b) \}$$

Describe all equivalence classes and show that they form a partition of $\{a, b, c, d\}$.

Operations on Binary Relations

A binary relation is a set $R \subseteq A \times B$.

It thus makes sense to consider set operations on binary relations.

Example: Consider the binary relations Mother, Father and Parent on a set of people. For example, $\mathrm{Mother}(x, y)$ indicates that x is mother of y. Then

- $\mathrm{Parent} = \mathrm{Father} \cup \mathrm{Mother}$
- $\mathrm{Father} = \mathrm{Parent} \setminus \mathrm{Mother}$

Exercise: Let R_1, R_2 and R_3 be given by

- $R_1 = \{ (x, y) \in \mathbb{N}^2 : x < y \}$ *(less-than)*
- $R_2 = \{ (x, y) \in \mathbb{N}^2 : x = y \}$ *(equality)*
- $R_3 = \{ (x, y) \in \mathbb{N}^2 : x \leq y \}$ *(less-than-or-equal-to)*

What are the following relations?

1. $R_1 \cup R_2$
2. $R_3 \cap \overline{R_2}$
3. $R_3 \setminus R_1$

Inverse Relation

A binary relation $R \subseteq A \times B$ can be turned around to give a relation $R^{-1} \subseteq B \times A$ by considering the converse relation:

$$R^{-1} = \{ (b, a) : (a, b) \in R \}.$$

R^{-1} is referred to as the ***inverse*** of R

Examples:

- The inverse of the *less-than-or-equal-to* relation $\leq$ is the *greater-than-or-equal-to* relation $\geq$:

$$(\leq)^{-1} = \geq \quad \textit{since}$$

$$x \leq y \text{ if, and only if, } y \geq x$$

- The inverse of the Parent relation is the Child relation:

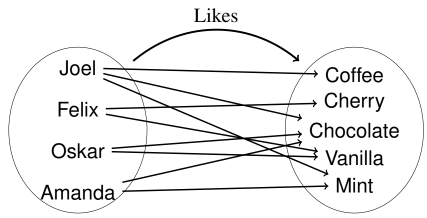
$$\text{Parent}^{-1} = \text{Child} \quad \textit{since}$$

$$x \text{ is parent of } y \text{ if, and only if, } y \text{ is child of } x$$

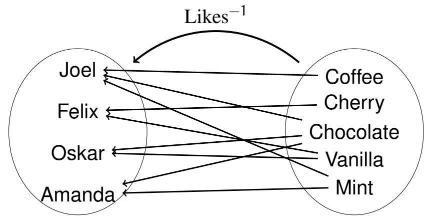
Exercise: What is the inverse of the Sibling relation?

Inverse Relations

The inverse of the relation $\text{Likes} \subseteq \text{Children} \times \text{Flavours}$



is the relation $\text{Likes}^{-1} \subseteq \text{Flavours} \times \text{Children}$



For example, $(\text{Joel}, \text{Mint}) \in \text{Likes}$ indicates that Joel *likes* mint ice cream, whilst $(\text{Mint}, \text{Joel}) \in \text{Likes}^{-1}$ indicates that mint ice cream *is liked by* Joel.

Functions as Relations

The graph of a function $f : A \to B$ is a binary relation $R_f \subseteq A \times B$ in which every $a \in A$ is related to exactly one $b \in B$:

$$a \, R_f \, b \text{ if, and only if, } f(a) = b.$$

Conversely, any binary relation $R \subseteq A \times B$ which satisfies this property is the graph of a function $f_R : A \to B$.

$$f_R(a) = b \text{ if, and only if, } aRb.$$

Given a relation $R \subseteq A \times B$:

- the ***domain*** of R is the set

$$\text{domain}(R) = \{ a \in A : \exists b \in B \, (aRb) \}$$

- the ***range*** of R is the set

$$\text{range}(R) = \{ b \in B : \exists a \in A \, (aRb) \}$$

Note: $\text{domain}(R_f) = \text{domain}(f)$ and $\text{range}(R_f) = \text{range}(f)$.

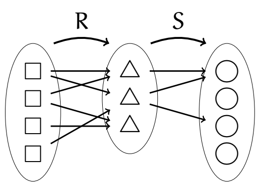
Example: Consider the relation Parent on humans:

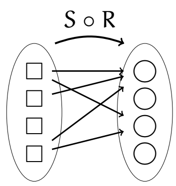
- $\text{domain}(\text{Parent})$ = set of all parents (*not* all humans)
- $\text{range}(\text{Parent})$ = set of all children (i.e., *all* humans)

Composing Relations

Suppose $R \subseteq A \times B$ and $S \subseteq B \times C$ are binary relations in which the sink of R is the same as the source of S.

In such an instance, we can compose these relations one after the other, much like we compose functions:





The ***composition*** of S and R is the relation $S \circ R \subseteq A \times C$ gotten by composing one relation after the other:

$$S \circ R = \{ (a, c) \in A \times C : \exists b \in B \, ((a, b) \in R \wedge (b, c) \in S) \}$$

Example: A grandfather is a father of a parent:

$$\text{Grandfather} = \text{Parent} \circ \text{Father}$$

Order matters: For example, a grandfather is a father of a parent, which is not the same as a parent of a father:

$$\text{Parent} \circ \text{Father} \neq \text{Father} \circ \text{Parent}$$