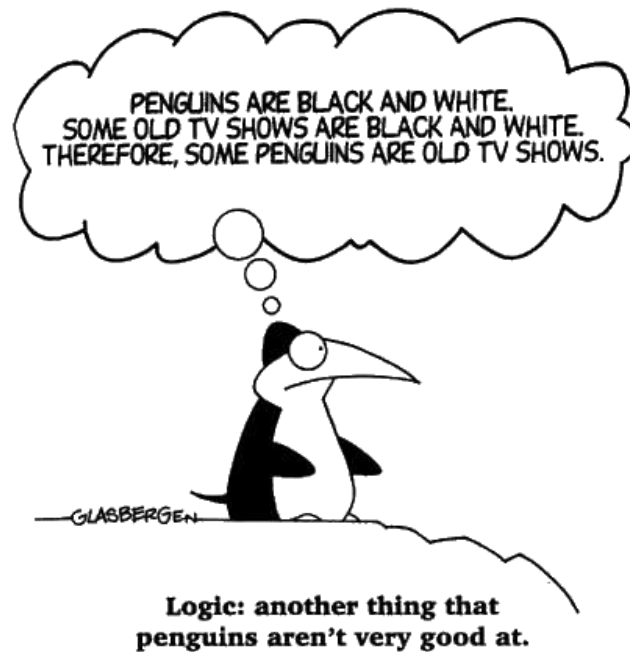


Reasoning & Logic (CSE1300) — Summary

Dany Sluijk

September/October 2018



Abstract

This document contains a summary of the Reasoning & Logic course given in the first year of Computer Science and Engineering. This is *not* a definitive guide, and might contain errors. Please send an email to “dany@atlasdev.nl”. A lot of information in this book is taken with fair use from the “Delftse Foundations of Computation” book created by Stefan Hugtenburg and Neil Yorke-Smith. This summary is distributed under the same CC BY-NC-SA 4.0 license.

Contents

| | | |
|----------|--|----------|
| 1 | Propositional Logic | 3 |
| 1.1 | Valid \neq Sound | 3 |
| 1.2 | Logical connectives | 3 |
| 1.3 | Logical equivalence | 4 |
| 1.4 | Sufficient or Necessary | 4 |
| 1.5 | Contrapositive, Converse and Inverse | 5 |
| 1.6 | Tautology, Contradiction and Contingency | 5 |

1 Propositional Logic

Propositional logic is about arguments. An argument contains two things:

- One or more premises.
- One conclusion.

Both of these things must be propositions. This means they must evaluate to either true (**1**) or false (**0**). Take the following example:

$$\begin{array}{l} \text{All humans are mortal} \\ \text{Socrates is human} \\ \hline \therefore \text{Socrates is mortal} \end{array}$$

Table 1: An example of an argument in English

In this example a premise “All humans are mortal” is given. This is something which can be either true or false. Given that both premises are true, we must accept the conclusion to also be true. In this case that means Socrates is a mortal. Propositional logic is not a one-size-fit-all solution. It does not have **quantifiers**, which means you cannot specify an amount of something. Propositional statements are generally not in english. The are replaces with **propositional variables** like p , q and r .

1.1 Valid \neq Sound

There is a difference in **sound** and **valid** arguments. A **valid** argument is an argument which is ‘syntactically’ valid. This means that the argument holds if all permises are true, but it does not mean the permises hold up. When the arguments hold up in the real world, then the argument is also **sound**. The given example is both valid as well as sound. If you change both occurances of ‘mortal’ to ‘immortal’ then the example is still valid, but no longer sound as the argument no longer holds up in the real world.

1.2 Logical connectives

Logical connectives or **logical operators** is the ‘glue’ between propositions. A list of possible logical connectives are shown in table 2. Logical connectives can be used to compare or combine multiple propositional variables. Take for example the following formula: $q \wedge p$. This takes two propositional variables p and q and compares that. This returns another boolean,

namely true when both p and q are true, or false in every other case. Using this knowledge you can create something called a **compound proposition**. This means it combines multiple proposition to a single, larger one. An example of a compound proposition is $(q \wedge p) \vee \neg r$. Both $q \wedge p$ and $\neg r$ are propositions, and combining gives a new one. If you exclude the parenthesis then the formula is parsed from left to right. So $q \wedge p \wedge r$ is equal to $(q \wedge p) \wedge r$ but not to $q \wedge (p \wedge r)$.

| Symbol | Name | Effect |
|------------|--------------|---|
| \wedge | conjunction | True when both propositions are true |
| \vee | disjunction | True when one or both propositions are true |
| \oplus | exclusive or | True when one of the propositions is true |
| \neg | negation | True when the proposition is false |
| \implies | implication | True if both sides are true or the left side is false |
| \iff | iff | True if both sides are true |

Table 2: A list of logical connectives

1.3 Logical equivalence

If you want to check if two propositions are equivalent (e.g. have the same value all the time), then you can use a **truth table**. A truth table is an table with all possible outcomes of a proposition. If two propositions have the outcome in the truth table then the two propositions are equivalent. take this truth table for example:

| q | p | $q \wedge p$ | $\neg(\neg q \vee \neg p)$ |
|-----|-----|--------------|----------------------------|
| 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 |
| 1 | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 |

As you can see, the two statements have the same values in the truth table, and are logically equivalent.

1.4 Sufficient or Necessary

Given is the following implication $p \implies q$. p is in this statement the **hypothesis**, and q is the **conclusion**. If the hypothesis is true then it's sufficient to say the conclusion is also true. If the conclusion is true then it's necessary to say the hypothesis is also true. It's impossible for the conclusion

to be false if the hypothesis is true. But when the hypothesis is false then the conclusion can be both true and false. So if the hypothesis is false then you can say that the statement holds.

1.5 Contrapositive, Converse and Inverse

Every implication has a contrapositive, converse and a inverse. They can, but don't have to be logically equivalent. Given the implication 'If this is Tuesday, then we are in Belgium' or in logic form $p \implies q$ we can get the contrapositive, converse and inverse of it. The **contrapositive** is $\neg q \implies \neg p$, so 'If we aren't in Belgium, then this isn't Tuesday'. The **converse** is $q \implies p$, so 'If we are in Belgium, then this is Tuesday'. The **inverse** is $\neg p \implies \neg q$, so 'If this isn't Tuesday, then we aren't in Belgium'.

1.6 Tautology, Contradiction and Contingency

There are more categories where propositions can fit in. These three are one of those. A tautology is a proposition which always evaluate to true. An example is the proposition $q \vee \neg q$. A contradiction is the opposite, it'll always evaluate to false. An example is $q \wedge \neg q$. A contingency is every thing else, which can be true or false. For example q .

| q | $q \vee \neg q$ | $q \wedge \neg q$ | q |
|-----|-----------------|-------------------|-----|
| 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 0 | 1 |