

Logic

M269 Unit 6

Contents

1	Agenda	2
2	Adobe Connect	3
2.1	Student View	3
2.2	Settings	4
2.3	Student & Tutor Views	6
2.4	Sharing Screen & Applications	8
2.5	Ending a Meeting	8
2.6	Invite Attendees	8
2.7	Layouts	9
2.8	Chat Pods	9
3	Introduction	9
3.1	Logic: Syntax, Semantics and Proof	11
4	Using Logical Equivalences	11
4.1	Logic Exs Quantifiers Q 1	15
4.2	Logic Exs Quantifiers Soln 1	15
4.3	Logic Exs Absorption Laws Q 1	16
4.4	Logic Exs Absorption Laws Soln 1	16
5	Truth Function	17
6	Using Logical Equivalences — Negation Exercises	21
6.1	Logic Exs Negation Qs	21
6.2	Logic Exs Negation Solns	22
6.3	Negation Exercises — Further Points	25
7	Interpretations for Predicate Logic	32
7.1	Logic Exs Interpretations Q 1	33
7.2	Logic Exs Interpretations Soln 1	33
8	Logical Arguments	34
9	Justified Arguments and Natural Deduction	35
9.1	Proofs in Tree Form	40
9.2	Self-Assessment activity 7.4	41
10	Calculating with Logic	42
10.1	Logic Puzzles — Introduction	42
10.2	Knights and Knaves	42
10.3	Knights and Knaves — Variant	43
10.4	Harder Logic Puzzles	44
10.5	Knights and Knaves — Answers	45
10.6	Knights and Knaves — Truth Table Answers	48

10.7 Harder Logic Puzzles — Solutions	51
11 Logic and Programming	55
12 Future Work	56
13 Web Sites & References	56
13.1 Web Sites	56
References	56

1 Agenda

- Welcome & Introductions
- Logic topics:
 - Propositional and predicate logic
 - Truth tables, logical equivalences and *valid* arguments
 - Truth and *interpretations* in logic
 - *Justified* arguments and *Natural Deduction*
- Exercises similar to CMAs and exam
- *Key aim*: Identify where people have problems and how to overcome them.
- Slides <http://www.pmo1yneux.co.uk/OU/M269FolderSync/M269TutorialNotes/M269TutorialLogic/>
- *Adobe Connect* — if you or I get cut off, wait till we reconnect (or send you an email)

Introductions — Me

- *Name* Phil Molyneux
- *Background* Physics and Maths, Operational Research, Computer Science
- *First programming languages* Fortran, BASIC, Pascal
- *Favourite Software*
 - Haskell — pure functional programming language
 - Text editors TextMate, Sublime Text — previously Emacs
 - Word processing in L^AT_EX
 - Mac OS X
- *Learning style* — I read the manual before using the software (really)

Introductions — You

- *Name* ?
- *Position in M269* ? Which part of which Units and/or Reader have you read ?

- *Particular topics you want to look at ?*
- *Learning Syle ?*

M269 Logic References

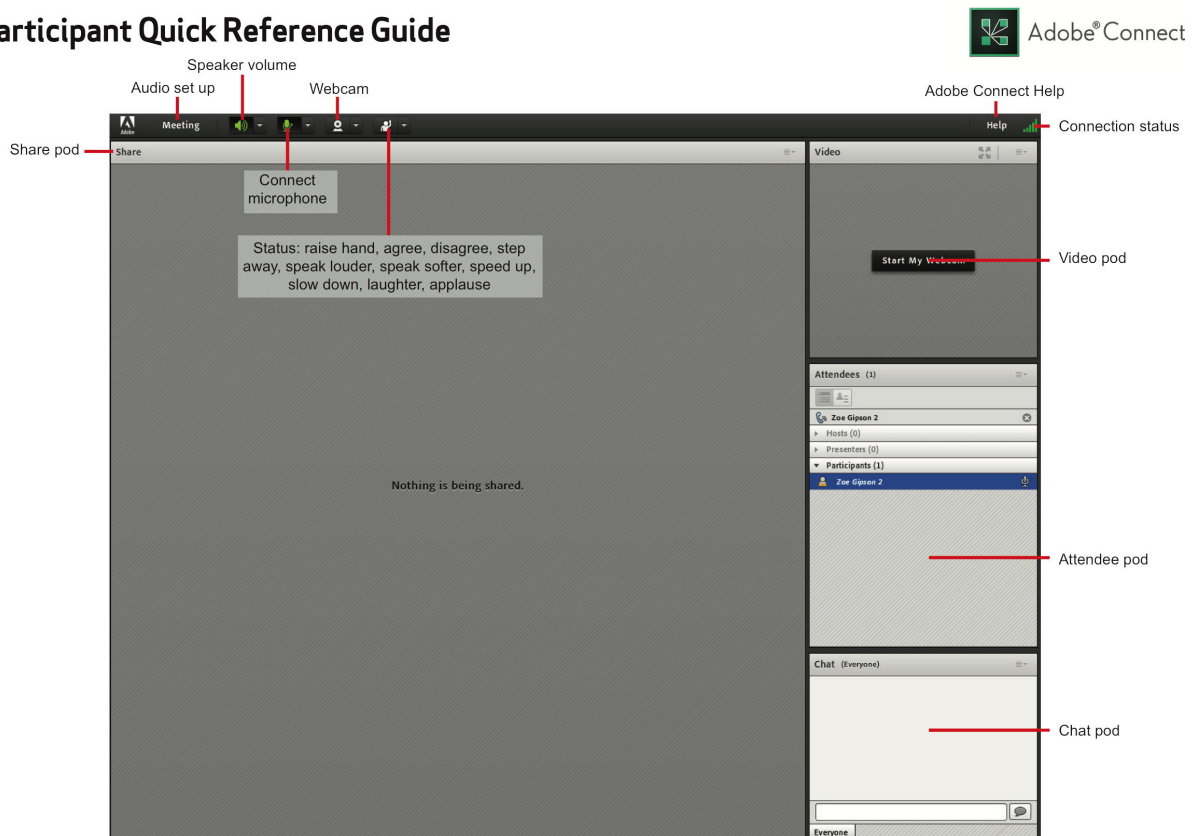
- M269 Unit 6 Section 1.2 Reading 6.2 — Chapter 2 of *Logic and the limits of computing, Propositional logic*
- M269 Unit 6 Section 2 Reading 6.3 — Chapter 3 of *Logic and the limits of computing, Relations and predicate logic*
- The above two introduce the idea of a *valid* argument
- M269 Unit 7 Section 2 *Logic revisited* — Section 2.3 *A proof system* introduces the idea of *justified* arguments and *Natural Deduction* proofs
- Material based on Allan Grimley's notes for M269 on *Natural Deduction*
- Calculating with logic — manipulating truth tables and finding equivalent propositions — logic puzzles (optional)

2 Adobe Connect Interface and Settings

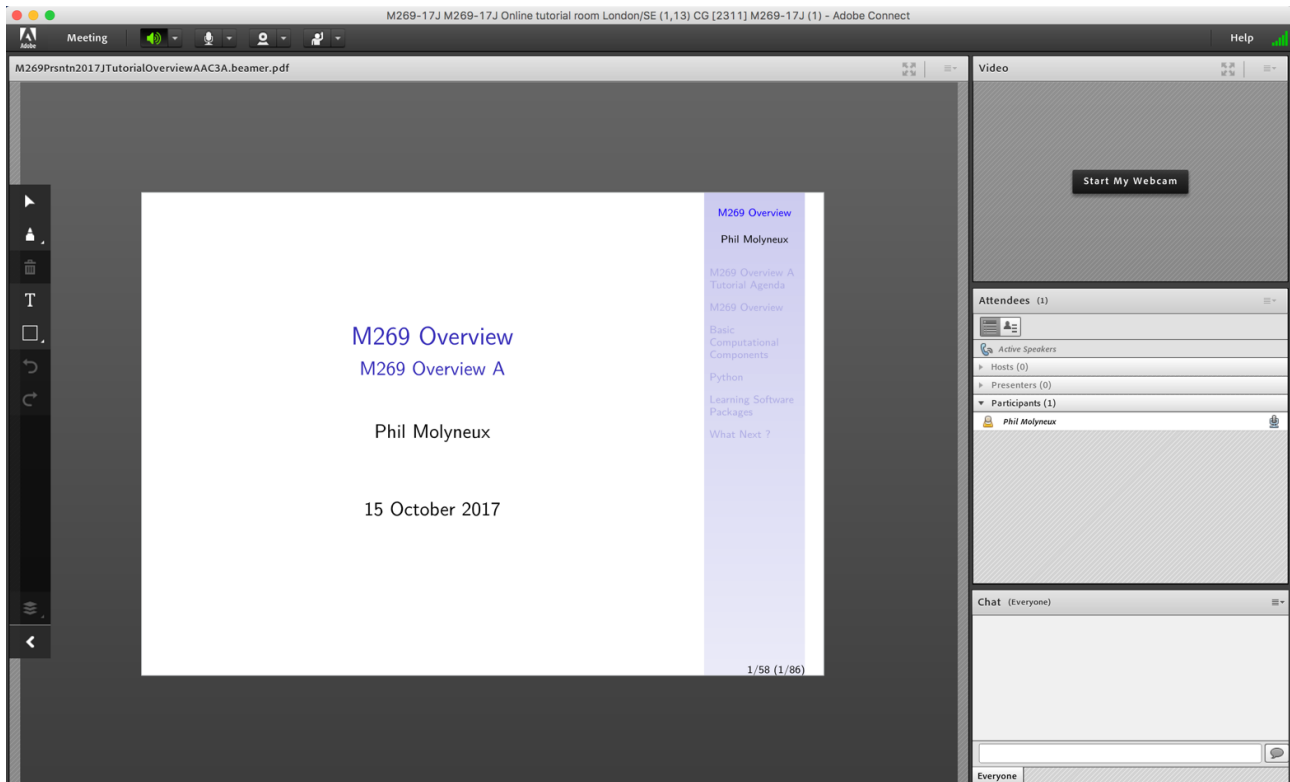
2.1 Adobe Connect Interface — Student View

Adobe Connect Interface — Student Quick Reference

Participant Quick Reference Guide



Adobe Connect Interface — Student View



2.2 Adobe Connect Settings

Adobe Connect Settings

- **Everybody: Audio Settings** Meeting > Audio Setup Wizard. . .
- **Audio** Menu bar > Audio > Microphone rights for Participants ✓
- Do not *Enable single speaker mode*
- **Drawing Tools** Share pod menu bar > Draw (1 slide/screen)
- Share pod menu bar > Menu icon > Enable Participants to draw ✓ gray
- Meeting > Preferences > Whiteboard > Enable Participants to draw ✓
- Cancel hand tool . . . Do not enable green pointer. . .
- Meeting > Preferences > Attendees Pod ✗ *Raise Hand notification*
- Meeting > Preferences > Display Name Display First & Last Name
- **Cursor** Meeting > Preferences > General tab > Host Cursors > Show to all attendees ✓ (default *Off*)
- Meeting > Preferences > Screen Share > Cursor > Show Application Cursor
- **Webcam** Menu bar > Webcam > Enable Webcam for Participants ✓
- **Recording** Meeting > Record Meeting. . . ✓

Adobe Connect — Access

- **Tutor Access**

TutorHome > M269 Website > Tutorials

Cluster Tutorials > M269 Online tutorial room

Tutor Groups > M269 Online tutor group room

Module-wide Tutorials > M269 Online module-wide room

- **Attendance**

TutorHome > Students > View your tutorial timetables

- **Beamer Slide Scaling 440% (422 x 563 mm)**

- **Clear Everyone's Status**

Attendee Pod > Menu > Clear Everyone's Status

- **Grant Access and send link via email**

Meeting > Manage Access & Entry > Invite Participants...

- **Presenter Only Area**

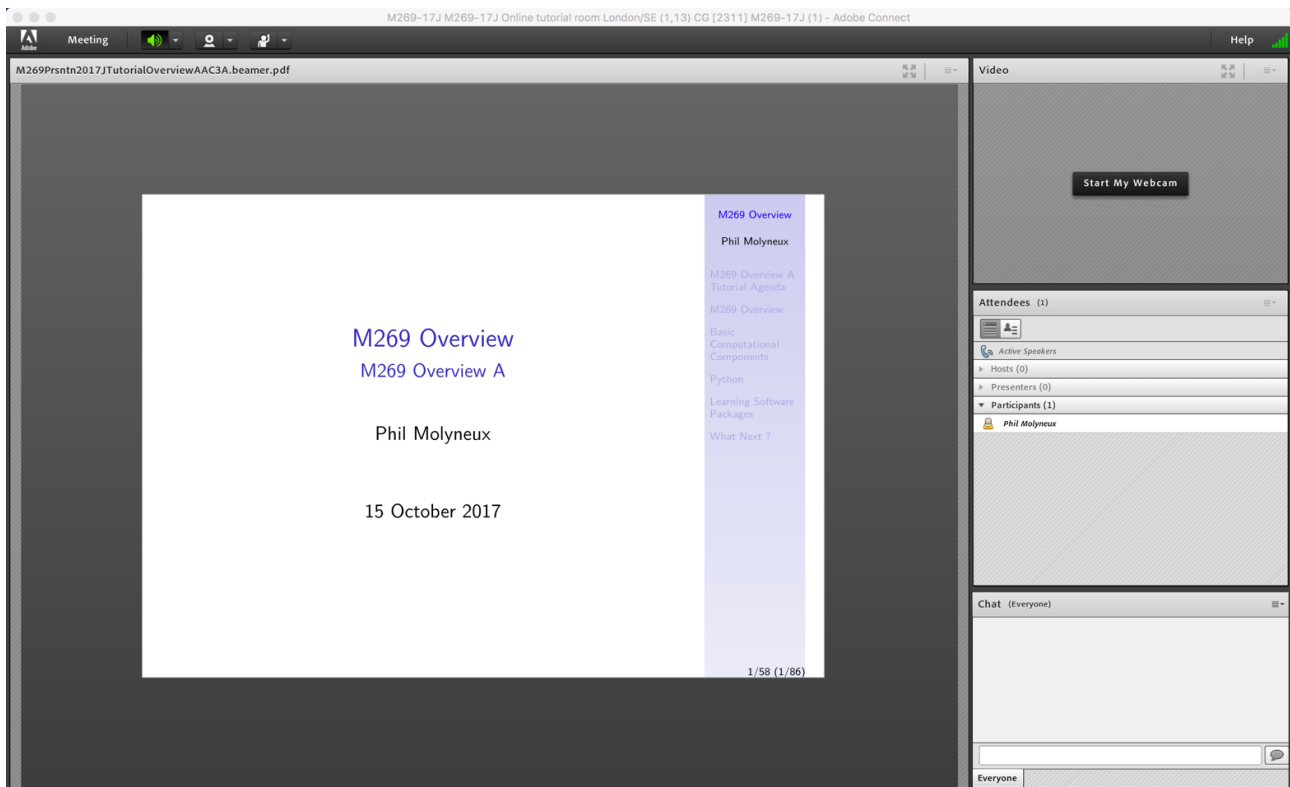
Meeting > Enable/Disable Presenter Only Area

Adobe Connect — Keystroke Shortcuts

- [Keyboard shortcuts in Adobe Connect](#)
- **Toggle Mic**  + **M** (Mac), **Ctrl** + **M** (Win) (On/Disconnect)
- **Toggle Raise-Hand status**  + **E**
- **Close dialog box**  (Mac), **Esc** (Win)
- **End meeting**  + 

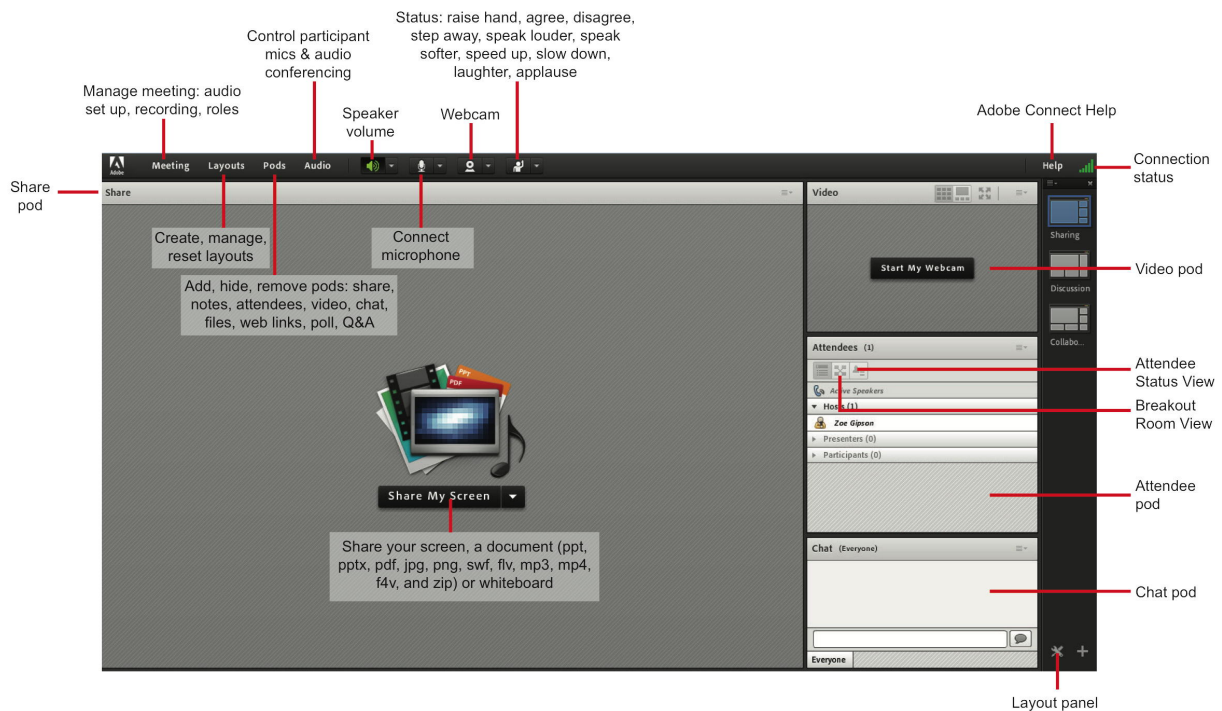
2.3 Adobe Connect Interface — Student & Tutor Views

Adobe Connect Interface — Student View (default)

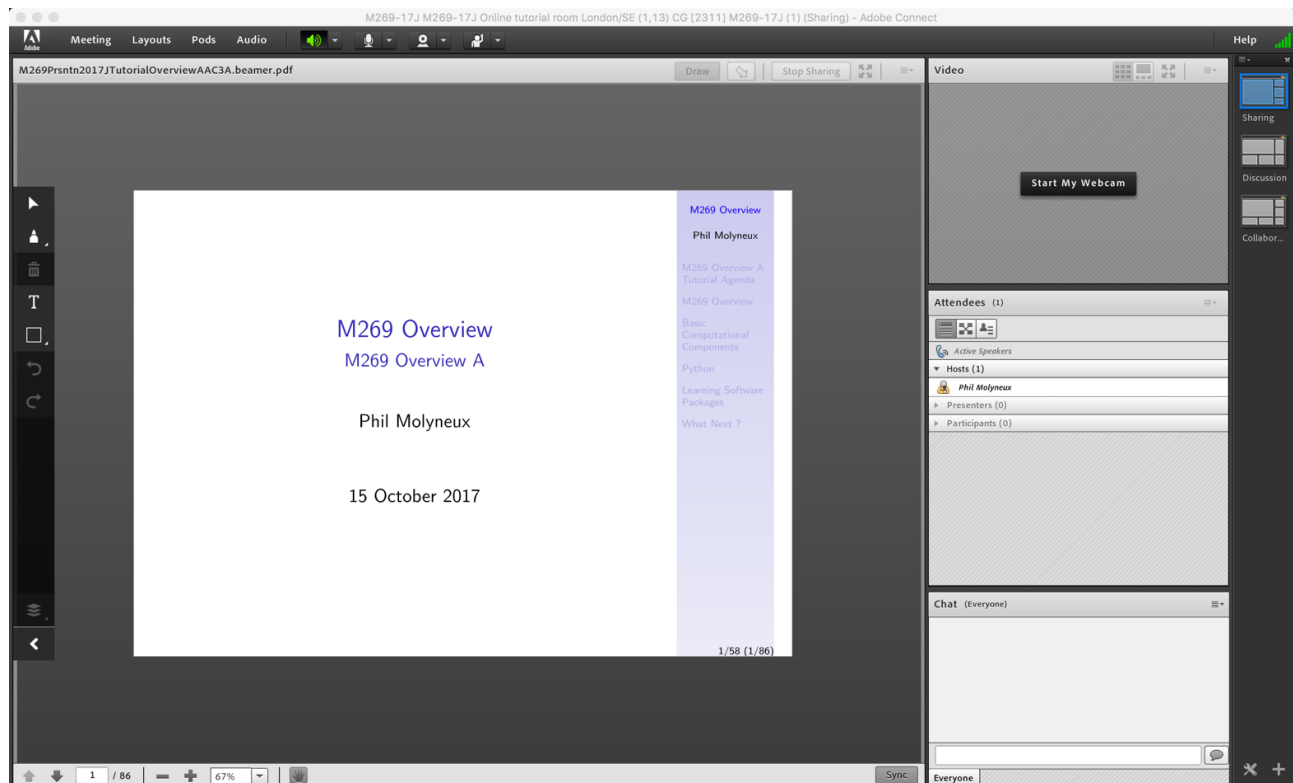


Adobe Connect Interface — Tutor Quick Reference

Host Quick Reference Guide



Adobe Connect Interface — Tutor View



2.4 Adobe Connect — Sharing Screen & Applications

- **Share My Screen** > **Application tab** > **Terminal** for **Terminal**
- **Share menu** > **Change View** > **Zoom in** for mismatch of screen size/resolution (Participants)
- (Presenter) Change to 75% and back to 100% (solves participants with smaller screen image overlap)
- Leave the application on the original display
- Beware blue hatched rectangles — from other (hidden) windows or contextual menus
- Presenter screen pointer affects viewer display — beware of moving the pointer away from the application
- First time: **System Preferences** > **Security & Privacy** > **Privacy** > **Accessibility**

2.5 Adobe Connect — Ending a Meeting

- *Notes for the tutor only*
- **Student:** **Meeting** > **Exit Adobe Connect**
- **Tutor:**
- **Recording** **Meeting** > **Stop Recording** ✓
- **Remove Participants** **Meeting** > **End Meeting. . .** ✓
 - Dialog box allows for message with default message:
 - *The host has ended this meeting. Thank you for attending.*
- **Recording availability** In course Web site for joining the room, click on the eye icon in the list of recordings under your recording — edit description and name
- **Meeting Information** **Meeting** > **Manage Meeting Information** — can access a range of information in Web page.
- **Attendance Report** see course Web site for joining room

2.6 Adobe Connect — Invite Attendees

- **Provide Meeting URL** **Menu** > **Meeting** > **Manage Access & Entry** > **Invite Participants. . .**
- **Allow Access without Dialog** **Menu** > **Meeting** > **Manage Meeting Information** provides new browser window with **Meeting Information** **Tab bar** > **Edit Information**
- Check *Anyone who has the URL for the meeting can enter the room*
- Default *Only registered users and accepted guests may enter the room*
- **Reverts to default next session but URL is fixed**
- Guests have blue icon top, registered participants have yellow icon top — same icon if URL is open

- See [Start, attend, and manage Adobe Connect meetings and sessions](#)

2.7 Layouts

- **Creating new layouts** example *Sharing* layout
- **Menu** > **Layouts** > **Create New Layout. . .** > **Create a New Layout dialog** > **Create a new blank layout** and name it *PMolyMain*
- New layout has no Pods but does have Layouts Bar open (see Layouts menu)
- **Pods**
- **Menu** > **Pods** > **Share** > **Add New Share** and resize/position — initial name is *Share n*
- **Rename Pod** **Menu** > **Pods** > **Manage Pods. . .** > **Manage Pods** > **Select** > **Rename** or **Double-click & rename**
- Add Video pod and resize/reposition
- Add Attendance pod and resize/reposition
- Add Chat pod — name it *PMolyChat* — and resize/reposition
- Dimensions of **Sharing** layout (on 27-inch iMac)
 - Width of Video, Attendees, Chat column 14 cm
 - Height of Video pod 9 cm
 - Height of Attendees pod 12 cm
 - Height of Chat pod 8 cm
- **Duplicating Layouts** does *not* give new instances of the Pods and is probably not a good idea (apart from local use to avoid delay in reloading Pods)

2.8 Chat Pods

- **Format Chat text**
- **Chat Pod** > **menu icon** > **My Chat Color**
- Choices: Red, Orange, Green, Brown, Purple, Pink, Blue, Black
- Note: Color reverts to Black if you switch layouts
- **Chat Pod** > **menu icon** > **Show Timestamps**

[Go to Table of Contents](#)

3 Introduction

- A plethora of logics, proof systems, and different notations can be puzzling.

- [Martin Davis, Logician](#) When I was a student, even the topologists regarded mathematical logicians as living in outer space. Today the connections between logic and computers are a matter of engineering practice at every level of computer organization

[Davis \(1995, page 289\)](#)

- Various logics, proof systems, were developed well before programming languages and with different motivations,

Mathematics and Notation

- **Richard Feynman** We could, of course, use any notation we want; do not laugh at notations; invent them, they are powerful. In fact, mathematics is, to a large extent, invention of better notations.

[Feynman et al. \(2011, The Feynman Lectures on Physics, 1963, Volume 1, chapter 17 Space-Time, section 17-5 Four-vector algebra\)](#)

- **Alfred North Whitehead** It is a profoundly erroneous truism, repeated by all copy-books and by eminent people when they are making speeches, that we should cultivate the habit of thinking of what we are doing. The precise opposite is the case. Civilization advances by extending the number of important operations which we can perform without thinking about them. Operations of thought are like cavalry charges in a battle — they are strictly limited in number, they require fresh horses, and must only be made at decisive moments.

[Whitehead \(1911, An Introduction to Mathematics, 1911, chapter 5\)](#)

Logic and Programming Languages

- Turing machines, [Von Neumann architecture](#) and procedural languages Fortran, C, Java, Perl, Python, JavaScript — [Hoare logic](#)
- [Resolution theorem proving](#) and [logic programming](#) — [Prolog](#)
- Logic and database query languages — SQL (Structured Query Language) and QBE (Query-By-Example) are syntactic sugar for first order logic
- [Lambda calculus](#) and functional programming with Miranda, Haskell, ML, Scala
- Programming languages are [formal systems](#) — that is, *specialized logics*
- [Hindley-Milner type system](#) a [type system](#) for the lambda calculus with parametric polymorphism — type system of [ML](#)
- [System F](#) also known as the (Girard-Reynolds) polymorphic lambda calculus or the second order lambda calculus — basis for languages such as [Haskell](#) and extensions in [GHC](#)

Reference: [Halpern et al. \(2001\)](#)

3.1 Logic: Syntax, Semantics and Proof

- The **syntax** of a logic defines the acceptable strings in the language — *well-formed formulae (WFFs)*
- The **semantics** of a logic associates meaning to a formula
- The **proof theory** is concerned with rules for manipulating formulae.
- Classical logic includes [Propositional logic](#) and [Predicate logic](#)
- [Propositional logic](#) has statements (or propositional constants) which can be True or False
 - It is raining
 - The assignment is due on Thursday
 - The exam is three hours long
- The statements (propositions) can be combined with logical connectives (functions of the propositions)
 - $\neg$ negation ($\neg p$)
 - $\wedge$ conjunction, AND ($p \wedge q$)
 - $\vee$ disjunction, OR ($p \vee q$)
 - $\Rightarrow$ logical implication, IF... THEN... ($p \Rightarrow q$)
 - Only expressions built from the rules are WFFs
- Proof systems including [Truth Tables](#) and [Natural Deduction](#)
- Note that there was a choice of connectives — see [Truth function](#) — the set given is [Functionally Complete](#) but is not minimal — see later
- [Predicate logic](#) uses quantified variables over sets and predicates indicating relations between objects.
- $\forall x.P(x)$ for all x , $P(x)$ is True
- $\exists x.Q(x)$ for some x , $Q(x)$ is True (or, there exists at least one x)
- Also called *first order logic*
- [Higher-order logic](#) quantifies over predicates, sets of sets, ... semantics more expressive but proof theories more complicated.

[Go to Table of Contents](#)

[Go to Table of Contents](#)

4 Using Logical Equivalences

- Unit 6 and chapters 2 and 3 of *Logic and the limits of computing* introduce *propositional* and *predicate* logic and some of the *equivalences* used in reasoning about statements.

- The following exercises ask you to prove the equivalence of some logic statements and the later exercises ask you to negate statements
- You can either think about them in English or translate them to statements in predicate logic and use the equivalences
- Which is the easiest ?
- And which is more reliable ?

Notation and Logical Equivalences

- We could define the notation for predicate calculus in a formal way and it is useful to eventually see we can make many of our definitions mechanical.
- At the start a formal definition can be intimidating until you have seen the usefulness of a formal approach.

Reference: See for example [Huth and Ryan \(2004, page 100\)](#)

It will be convenient to drop the set in the *binding term* of a quantifier when the set is obvious or irrelevant and we want to make an expression less cluttered. So most of this note will have $\forall x[p]$ and $\exists x[p]$ — the square brackets still denote the *scope* of the meaning of the variable name introduced by the quantifier.

If we were defining the language formally, we would also have to specify operator precedence and associativity (as in plain ordinary arithmetic: multiplication before addition and subtraction is left associative).

Formula	
$P(t_1, t_2, \dots, t_n)$	Predicate with arguments
$\neg p$	Negation of formula p
$\forall x \text{ in } X[p]$	Universal quantification
$\exists x \text{ in } X[p]$	Existential quantification
$p \wedge q$	Logical AND, conjunction
$p \vee q$	Logical OR, disjunction
$p \Rightarrow q$	Logical implication
(p)	Brackets

- Truth tables define the meaning of $\neg, \wedge, \vee, \Rightarrow$

Truth Tables for $\neg, \wedge, \vee, \Rightarrow$

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

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

- **Exercise** Justify the truth table for $\Rightarrow$

Justification of Truth Table for $\Rightarrow$

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

- The **True** values in the last two rows give students a lot of trouble
- What is going on ? This is a *negative definition*
- $p \Rightarrow q$ holds unless we have evidence to the contrary
- $\Rightarrow$ is one of the 16 possible truth functions of two boolean inputs
- In a typed programming language
- $\Rightarrow :: (\mathbb{B}, \mathbb{B}) \rightarrow \mathbb{B}$

Logical Equivalences

Reference: [Tunncliffe \(1991\)](http://en.wikipedia.org/wiki/Logical_equivalence), http://en.wikipedia.org/wiki/Logical_equivalence

Negation and De Morgan

Negation

$$p \vee \neg p \equiv \text{True}$$

$$p \wedge \neg p \equiv \text{False}$$

$$\neg \neg p \equiv p$$

De Morgan

$$\neg(p \vee q) \equiv \neg p \wedge \neg q$$

$$\neg(p \wedge q) \equiv \neg p \vee \neg q$$

$$\neg \forall x[P(x)] \equiv \exists x[\neg P(x)]$$

$$\neg \exists x[P(x)] \equiv \forall x[\neg P(x)]$$

- **Question** Why has the author put the equivalence symbol ($\equiv$) in a different colour ?
- The equivalence symbol ($\equiv$) is not a symbol in Propositional or Predicate Logic (in our notation)

- It is important to realise we have some notation to refer to notation in Logic
- This is common when we have proofs about logical statements
- Sadly most texts just use black and white
- And I haven't had time to do consistent colour coding (and would have to hack the package used for the proof tree layout)

De Morgan in Alice in Wonderland

- *White King* ... Just look along the road, and tell me if you can see either of them.
- *Alice* I can see nobody on the road
- *White King* ... To be able to see Nobody! And at that distance too!
- *Through the Looking Glass and What Alice Found There* Chp 7 *The Lion and the Unicorn*
- What was the day job of *Lewis Carroll* ?

Logical Equivalences

Rewriting $\Rightarrow$

Rewriting $\Rightarrow$

$$p \Rightarrow q \equiv \neg p \vee q$$

$$p \Leftrightarrow q \equiv (p \Rightarrow q) \wedge (q \Rightarrow p)$$

- **Exercise** Use a truth table to prove $p \Rightarrow q \equiv \neg p \vee q$

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

Logical Equivalences

$\vee, \wedge$ Laws

Distributive Laws

$$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$$

$$p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$$

Associative Laws

$$p \vee (q \vee r) \equiv (p \vee q) \vee r$$

$$p \wedge (q \wedge r) \equiv (p \wedge q) \wedge r$$

Commutative Laws

$$p \vee q \equiv q \vee p$$

$$p \wedge q \equiv q \wedge p$$

Logical Equivalences**Extended Commutativity****Extended Commutativity**

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

$$\exists x[\exists y[P(x, y)]] \equiv \exists y[\exists x[P(x, y)]]$$

often written $\forall x, \forall y[P(x, y)]$

often written $\exists x, \exists y[P(x, y)]$

4.1 Logic Exs Quantifiers Q 1

- Is it the case that $\exists x[\forall y[P(x, y)]] \equiv \forall y[\exists x[P(x, y)]]$?
- If not, give counter examples.
- Does $\forall y[\exists x[P(x, y)]] \Rightarrow \exists x[\forall y[P(x, y)]]$
- or does $\exists x[\forall y[P(x, y)]] \Rightarrow \forall y[\exists x[P(x, y)]]$

[Go to Quantifiers Soln 1](#)

[Go to Table of Contents](#)

4.2 Logic Exs Quantifiers Soln 1

- It is a common error to think they are equivalent
- See [Maths Stack Exchange: Is \$\forall x \exists y Q\(x, y\)\$ the same as \$\exists y \forall x Q\(x, y\)\$?](#)
- See [Maths Stack Exchange: What does \$\forall x \exists y \(x + y = 0\)\$ mean ?](#)
- Let $P(x, y)$ be $x + y = 0$
- Then $\forall x[\exists y[P(x, y)]]$ is true — say this in English
- but $\exists y[\forall x[P(x, y)]]$ is not true

[Go to Quantifiers Q 1](#)

[Go to Table of Contents](#)

Logical Equivalences

Other Equivalences

Identity Laws

$$p \vee \text{False} \equiv p$$

$$p \wedge \text{True} \equiv p$$

$$p \vee \text{True} \equiv \text{True}$$

$$p \wedge \text{False} \equiv \text{False}$$

Idempotent Laws

$$p \vee p \equiv p$$

$$p \wedge p \equiv p$$

Absorption Laws

$$p \vee (p \wedge q) \equiv p$$

$$p \wedge (p \vee q) \equiv p$$

4.3 Logic Exs Absorption Laws Q 1

- Prove the *Absorption Laws* using truth tables
- Prove the *Absorption Laws* using other equivalences

[Go to Absorption Laws Soln 1](#)

[Go to Table of Contents](#)

4.4 Logic Exs Absorption Laws Soln 1

- Truth table for $p \vee (p \wedge q) \equiv p$

p	q	$p \wedge q$	$p \vee (p \wedge q)$
T	T	T	T
T	F	F	T
F	T	F	F
F	F	F	F

[Go to Absorption Laws Q 1](#)

- Equivalences proof for $p \vee (p \wedge q) \equiv p$
- $p \vee (p \wedge q)$
- $\rightarrow (p \vee p) \wedge (p \vee q)$ by *Distributive laws*
- $\rightarrow p \wedge (p \vee q)$ by *Idempotent laws*
- This could go round in circles — *start again*.

[Go to Absorption Laws Q 1](#)

- Equivalences proof for $p \vee (p \wedge q) \equiv p$
- $p \vee (p \wedge q)$
- $\rightarrow (p \wedge T) \vee (p \wedge q)$ by *Identity laws Eureka step*
- $\rightarrow p \wedge (T \vee q)$ by *Distributive laws*
- $\rightarrow p \wedge T$ by *Identity & Commutative laws*
- $\rightarrow p$ by *Identity laws*

[Go to Absorption Laws Q 1](#)

[Go to Table of Contents](#)

[Go to Table of Contents](#)

5 Truth Function

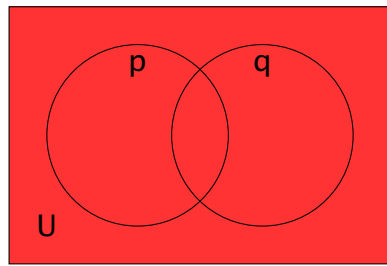
- The following notes illustrate the 16 binary functions of two Boolean variables
- See [Truth function](#)
- See [Functional completeness](#)
- See [Sheffer stroke](#)
- See [Logical NOR](#)

Table of Binary Truth Functions

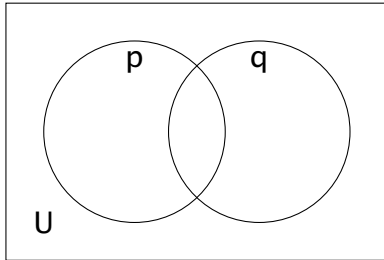
p	q	T	$p > q$	$p \downarrow q$	$p \uparrow q$	$p \leftrightarrow q$	$p \oplus q$	$p < q$
T	T	T	T	T	T	T	T	T
T	F	T	T	T	F	F	F	F
F	T	T	T	F	T	T	F	F
F	F	T	F	T	F	F	T	F

p	q	$\perp$	$p \leq q$	$p \nabla q$	$p \vdash q$	$p \dashv q$	$p \boxplus q$	$p \boxless q$
T	T	F	F	F	F	F	F	F
T	F	F	F	F	T	T	T	T
F	T	F	F	T	T	F	T	T
F	F	F	T	F	T	F	F	T

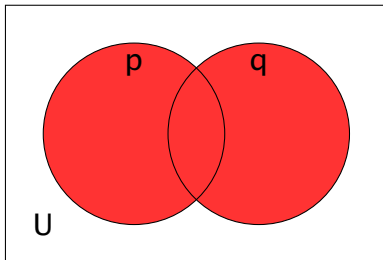
- [Tautology](#) True, $\top$, *Top*



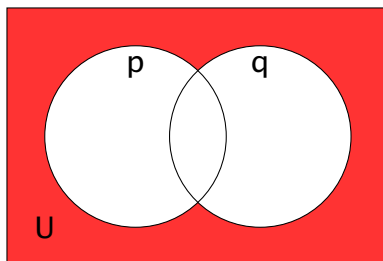
- **Contradiction** False, $\perp$, *Bottom*



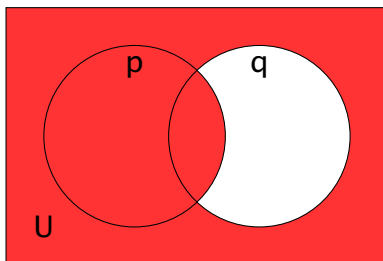
- **Disjunction OR**, $p \vee q$



- **Joint Denial NOR**, $p \supseteq q$, $p \downarrow q$, *Pierce's arrow*



- **Converse Implication** $p \Leftarrow q$



- **Converse Nonimplication** $p \nLeftarrow q$

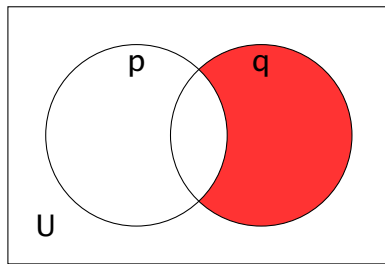
p	q	$\perp$
T	T	T
T	F	T
F	T	T
F	F	T

p	q	$\perp$
T	T	F
T	F	F
F	T	F
F	F	F

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

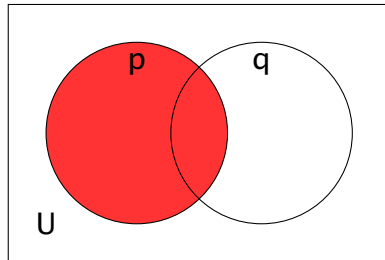
p	q	$p \downarrow q$
T	T	F
T	F	F
F	T	F
F	F	T

p	q	$p \Leftarrow q$
T	T	T
T	F	T
F	T	F
F	F	T



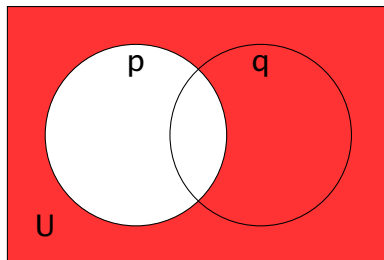
p	q	$p \nrightarrow q$
T	T	F
T	F	F
F	T	T
F	F	F

- Proposition p



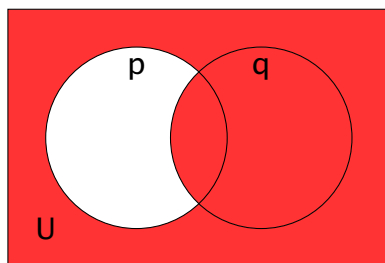
p	q	p
T	T	T
T	F	T
F	T	F
F	F	F

- Negation of p $\neg p$



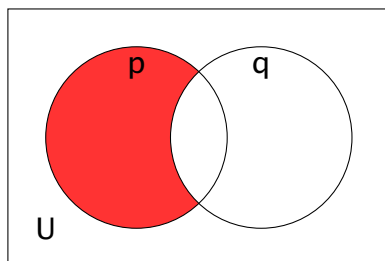
p	q	$\neg p$
T	T	F
T	F	F
F	T	T
F	F	T

- Material Implication $p \Rightarrow q$



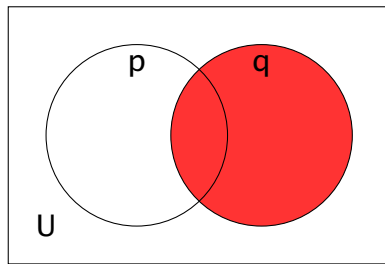
p	q	$p \Rightarrow q$
T	T	T
T	F	F
F	T	T
F	F	T

- Material Nonimplication $p \nRightarrow q$



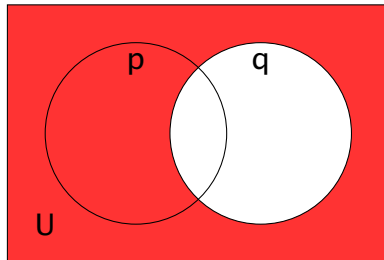
p	q	$p \nRightarrow q$
T	T	F
T	F	T
F	T	F
F	F	F

- Proposition q



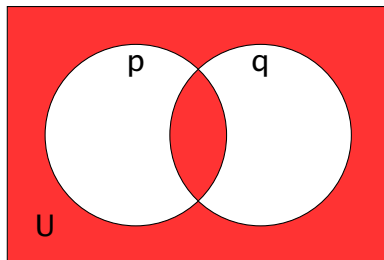
p	q	$p \rightarrow q$
T	T	T
T	F	F
F	T	T
F	F	T

- **Negation of q** $\neg q$



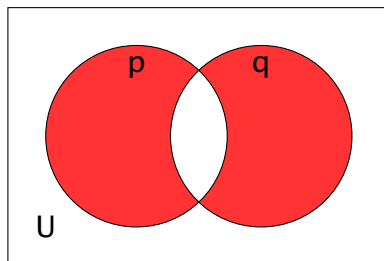
p	q	$\neg q$
T	T	F
T	F	T
F	T	F
F	F	T

- **Biconditional** If and only if, IFF, $p \Leftrightarrow q$



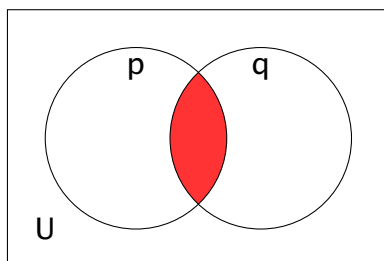
p	q	$p \Leftrightarrow q$
T	T	T
T	F	F
F	T	F
F	F	T

- **Exclusive disjunction XOR**, $p \oplus q$, $p \vee q$



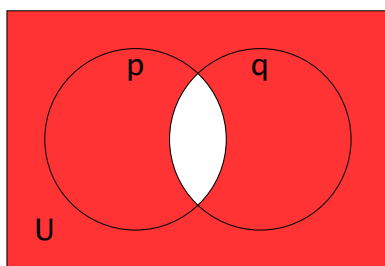
p	q	$p \oplus q$
T	T	F
T	F	T
F	T	T
F	F	F

- **Conjunction AND**, $p \wedge q$



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

- **Alternative denial NAND**, $p \bar{\wedge} q$, $p \uparrow q$, *Sheffer stroke*



p	q	$p \uparrow q$
T	T	F
T	F	T
F	T	T
F	F	T

- **Functionally complete** set of connectives is one which can be used to express all possible connectives
- $p \Rightarrow q \equiv \neg p \vee q$ so we could just use $\{\neg, \wedge, \vee\}$
- **Boolean programming** — we have to have a functionally complete set but choose more to make the programming easier
- *Expressiveness* is an issue in programming language design
- **NAND** $p \bar{\wedge} q$, $p \uparrow q$, Sheffer stroke
- **NOR** $p \bar{\vee} q$, $p \downarrow q$, Pierce's arrow
- Both $\{\uparrow\}, \{\downarrow\}$ are functionally complete — verify:

$$\begin{aligned}
 \neg p &\equiv p \uparrow p \\
 p \wedge q &\equiv \neg(p \uparrow q) \equiv (p \uparrow q) \uparrow (p \uparrow q) \\
 p \vee q &\equiv (p \uparrow p) \uparrow (q \uparrow q) \\
 p \Rightarrow q &\equiv ((p \uparrow p) \uparrow (p \uparrow p)) \uparrow (q \uparrow q) \\
 \neg p &\equiv p \downarrow p \\
 p \wedge q &\equiv (p \downarrow p) \downarrow (q \downarrow q) \\
 p \vee q &\equiv \neg(p \downarrow q) \equiv (p \downarrow q) \downarrow (p \downarrow q) \\
 p \Rightarrow q &\equiv ((p \downarrow p) \downarrow q) \downarrow ((p \downarrow p) \downarrow q)
 \end{aligned}$$

- Not a novelty — the [Apollo Guidance Computer](#) was implemented in **NOR** gates alone.

[Go to Table of Contents](#)

6 Using Logical Equivalences — Negation Exercises

6.1 Logic Exs Negation Qs

- In each of the following questions $P(x, y, \dots)$ denotes a statement involving objects $x, y, \dots$. Construct the negation of each of the following propositions.
 1. $P(x)$ is true for all x .
 2. $P(x, y)$ is true for all x and all y .
 3. There is at least one x such that $P(x, y)$ is true for all y .
 4. Given any x there is at least one y such that $P(x, y)$ is false.

5. Given any x there is at least one y such that $P(x, y, z)$ is true for all z .
6. Given any x there is precisely one y such that $P(x, y, z)$ is true for at least one z .
7. Given any x there is at least one y such that $P(x, y, z)$ is true for at most one z .

[Go to Negation Solns](#)

[Go to Table of Contents](#)

6.2 Logic Exs Negation Solns

- Our strategy:
- Translate the English statements into our *formal language*
- Use the equivalence rules to simplify the negation
- Finally translate back into English

[Go to Negation Qs](#)

Using Logical Equivalences — Answers

Using Logical Equivalences — Answer 1

- Our strategy is to translate the English statements into our formal language, use the equivalence rules to simplify the negation and finally translate back to English
- $P(x)$ is true for all x .
- **Translate** $\forall x[P(x)]$
- **Negation** $\neg(\forall x[P(x)])$
- **Simplify** $\exists x[\neg P(x)]$
- **Translate** $P(x)$ is false for at least one x

[Go to Negation Qs](#)

Using Logical Equivalences — Answer 2

- $P(x, y)$ is true for all x and all y .
- **Translate** $\forall x, \forall y[P(x, y)]$
- **Negation** $\neg(\forall x, \forall y[P(x, y)])$
- **Simplify** $\exists x, \exists y[\neg P(x, y)]$
- **Translate** $P(x, y)$ is false for at least one x and one y

[Go to Negation Qs](#)

Using Logical Equivalences — Answer 3

- There is at least one x such that $P(x, y)$ is true for all y .
- **Translate** $\exists x[\forall y[P(x, y)]]$
- **Negation** $\neg(\exists x[\forall y[P(x, y)]])$
- **Simplify** $\forall x[\exists y[\neg P(x, y)]]$
- **Translate** Given any x there is at least one y (possibly depending on x) such that $P(x, y)$ is false

[Go to Negation Qs](#)**Using Logical Equivalences — Answer 4**

- Given any x there is at least one y such that $P(x, y)$ is false.
- **Translate** $\forall x[\exists y[\neg P(x, y)]]$
- **Negation** $\neg(\forall x[\exists y[\neg P(x, y)]])$
- **Simplify** $\exists x[\forall y[P(x, y)]]$
- **Translate** There is at least one x such that for all y , $P(x, y)$ is true.

[Go to Negation Qs](#)**Using Logical Equivalences — Answer 5**

- Given any x there is at least one y such that $P(x, y, z)$ is true for all z .
- **Translate** $\forall x[\exists y[\forall z[P(x, y, z)]]]$
- **Negation** $\neg(\forall x[\exists y[\forall z[P(x, y, z)]]])$
- **Simplify** $\exists x[\forall y[\exists z[\neg P(x, y, z)]]]$
- **Translate** There is at least one x such that for all y there is at least one z (possibly depending on y) such that $P(x, y, z)$ is false.

[Go to Negation Qs](#)**Using Logical Equivalences — Answer 6**

- Given any x there is at precisely one y such that $P(x, y, z)$ is true for at least one z .
- **Translate** $\forall x[\exists!y[\exists z[P(x, y, z)]]]$ *Note $\exists!$ for exactly one*
- **Eureka Step** *Exactly one means At least one and not two or more*
- **Expand** $\forall x[\exists y[\exists z[P(x, y, z)]]$
 $\wedge$
 $\neg(\exists y_1, \exists y_2[y_1 \neq y_2 \wedge \exists z[P(x, y_1, z)] \wedge \exists z[P(x, y_2, z)]])]$
- **Negation** $\neg(\forall x[\exists y[\exists z[P(x, y, z)]]$
 $\wedge$
 $\neg(\exists y_1, \exists y_2[y_1 \neq y_2 \wedge \exists z[P(x, y_1, z)] \wedge \exists z[P(x, y_2, z)]]))$

- **Simplify** $\exists x[\forall y[\forall z[\neg P(x, y, z)]]]$
 $\vee$
 $(\exists y_1, \exists y_2[y_1 \neq y_2 \wedge \exists z[P(x, y_1, z)] \wedge \exists z[P(x, y_2, z)]])]$
- **Translate** For at least one x there is either no y and z such that $P(x, y, z)$ is true or there are at least two y such that there exists a z (possible depending on the y) such that $P(x, y, z)$ is true.

[Go to Negation Qs](#)

Using Logical Equivalences — Answer 7

- Given any x there is at least one y such that $P(x, y, z)$ is true for at most one z .
- **Translate** $\forall x[\exists y[\text{for at most one } z[P(x, y, z)]]]$
- *Note lack of notation here*
- **Eureka Step** *At most one* means *none or exactly one* (we will have a lot of code here)
- **Expand** $\forall x[\exists y[\neg \exists z[P(x, y, z)]]]$
 $\vee$
 $(\exists z[P(x, y, z)]$
 $\wedge$
 $\neg(\exists z_1, \exists z_2[z_1 \neq z_2 \wedge P(x, y, z_1) \wedge P(x, y, z_2)]))]$
- **Negation** $\neg(\forall x[\exists y[\neg \exists z[P(x, y, z)]]]$
 $\vee$
 $(\exists z[P(x, y, z)]$
 $\wedge \neg(\exists z_1, \exists z_2[z_1 \neq z_2 \wedge P(x, y, z_1) \wedge P(x, y, z_2)]))]$
- **Simplify** $\exists x[\forall y[\exists z[P(x, y, z)]]]$
 $\wedge$
 $(\forall z[\neg P(x, y, z)]$
 $\vee (\exists z_1, \exists z_2[z_1 \neq z_2 \wedge P(x, y, z_1) \wedge P(x, y, z_2)]))]$
- **Simplify back up** $\exists x[\forall y[\exists z[P(x, y, z)]]]$
 $\wedge$
 $(\neg \exists z[P(x, y, z)]$
 $\vee (\exists z_1, \exists z_2[z_1 \neq z_2 \wedge P(x, y, z_1) \wedge P(x, y, z_2)]))]$
- Now use the *Distributive Law*
- **Distributive Law** $\exists x[\forall y[(\exists z[P(x, y, z)] \wedge \neg \exists z[P(x, y, z)])]]]$
 $\vee$
 $(\exists z[P(x, y, z)]$
 $\wedge (\exists z_1, \exists z_2[z_1 \neq z_2 \wedge P(x, y, z_1) \wedge P(x, y, z_2)]))]$
- Now use the *Negation Law*
- **Negation Law** $\exists x[\forall y[$
 (False)
 $\vee$
 $(\exists z[P(x, y, z)]$
 $\wedge (\exists z_1, \exists z_2[z_1 \neq z_2 \wedge P(x, y, z_1) \wedge P(x, y, z_2)]))]$

- **Absorption Law** $\exists x[\forall y[\exists z[P(x, y, z) \wedge (\exists z_1, \exists z_2[z_1 \neq z_2 \wedge P(x, y, z_1) \wedge P(x, y, z_2)])]]]$
- **Translate** There exists at least one x such that for all y , $P(x, y, z)$ is true for more than one z

[Go to Negation Qs](#)

Logical Equivalences — Comments on Exercises

- *Plain English* is never that plain
- Consider: *Fruit flies like a banana*
- A good notation should help clarify thought — see Whitehead quote
- Note how the ordering of clauses in English can lead to ambiguity — does a z depend on a previous y , for example — hence we need a precisely defined notation to determine scope of variables
- Using a *formal language* can help the manipulation but there is no free lunch
- You need a decent editor to check your syntax and bracket matching — software exists to help this — see [Wikipedia Proof Assistant](#)

[Go to Negation Qs](#)

[Go to Table of Contents](#)

6.3 Negation Exercises — Further Points

- The above exercises were just about the only instruction on Propositional and Predicate Logic I had as an undergraduate (in Physics and Maths, Sussex University)
- Below are copies of the original question sheet and my answers with markers comments.
- Notice that my mistakes mainly involved getting the order of the English clauses wrong — in English, it is harder to *see* the scope of names.
- I also confused the colloquial *at least one x or at least one y* for *at least one x and at least one y*

Negation Exercises — Further Points Exs (1)

1 Oct

ANALYSIS (Mathematics subject 4a)DEFINITIONS

1. If P and Q are propositions (which may be true or false) we say that ' P implies Q ' (in symbols $P \Rightarrow Q$) if the truth of P ensures the truth of Q .

Alternatively we have, a priori, the four possibilities:

- (i) P and Q are both true
- (ii) P and Q are both false
- (iii) P is false and Q is true
- (iv) P is true and Q is false

' P implies Q ' means that the fourth alternative, but only the fourth, is excluded.

The point here is that if P is false ' P implies Q ' provides no information whatever about (or imposes no restriction on) Q .

11. If ' $P \Rightarrow Q$ ' and ' $Q \Rightarrow P$ ' we say that ' P is equivalent to Q ' and write $P \Leftrightarrow Q$.

111. The statement 'The proposition Q is false' is called the negation of the statement 'The proposition Q is true'.

Reference

Scott, D.B. and Tims, S.R. Mathematical Analysis: An introduction
 §0.2 p3 - 9, C.U.P 60/-
 (also recommended for the Analysis (4a) course)

Negation Exercises — Further Points Exs (2)

Exercises (to be attempted in Registration week)

In each of the following questions two conditions are given. Decide in each case whether condition (a) is necessary or sufficient (or both) or neither, for (b), and then answer the same question with (a) and (b) interchanged. The usual notation for a triangle is employed in questions 1, 2, 3 and 8.

1. (a) The angle A is obtuse
(b) a exceeds each of b and c.
2. (a) A exceeds $\pi/3$
(b) a exceeds each of b and c.
3. (a) a^2 exceeds b^2+c^2
(b) A is obtuse.
4. (a) $x^2-3x+2 = 0$
(b) $x=1$.
5. (a) $\frac{y_1}{x_1} = \frac{y_2}{x_2} = \frac{y_3}{x_3}$ and $x_1x_2x_3 \neq 0$.
(b) the three points (x_1, y_1) , (x_2, y_2) , (x_3, y_3) are collinear.

In each of the preceding five and the next three following questions does (a) imply (b) or conversely does (b) imply (a)?

6. (a) x exceeds 2
(b) x exceeds 1.
7. (a) $x^4-5x^2+4 = 0$
(b) $x=1$ or $x=2$.
8. (a) $a > b+c$
(b) A is obtuse.

Negation Exercises — Further Points Exs (3)

- 2 -

In each of the following questions $P(x, y, \dots)$ denotes a statement involving objects $x, y, \dots$. Construct the negation of each of the following propositions.

9. $P(x)$ is true for all x .
10. $P(x, y)$ is true for all x and all y .
11. There is at least one x such that $P(x, y)$ is true for all y .
12. Given any x there is at least one y such that $P(x, y)$ is false.
- * 13. Given any x there is at least one y such that $P(x, y, s)$ is true for all s .
- * 14. Given any x (there is precisely one y such that $P(x, y, s)$ is true for at least one s .)
- * 15. Given any x (there is at least one y such that $P(x, y, s)$ is true for at most one s .)

$\exists x$ s.t. given any y $P(x, y, z)$ is ^{true} ~~false~~ for at least two z

(* more difficult question)

Negation Exercises — Further Points Answers (1)

SP. MOLYNEUX 2 Oct 1968
ANALYSIS (MATHS 4A)

1. a) is sufficient for b) ✓
 b) is necessary for a) ✓
 a) implies b) b) does not imply a)
2. a) is necessary for b) ✓
 b) is sufficient for a) ✓
 b) implies a) a) does not imply b)
3. a) is both necessary and sufficient for b) ✓
 b) " " " " " " a) ✓
 a) is equivalent to b)
4. a) is necessary for b) ✓
 b) is sufficient for a) ✓
 b) implies a) a) does not imply b)
5. a) is sufficient for b) ✓
 b) is necessary for a) ✓
 a) implies b) b) does not imply a)
6. a) implies b) ✓
7. b) implies a) ✓ (solution of a) is $x = \frac{\pm 1}{1}$ or ± 2)
8. a) implies b) ✓

Negation Exercises — Further Points Answers (2)

The negations are

9. $P(x)$ is false for at least one x .
10. $P(x, y)$ is false for at least one x ^{and} one y .
11. There is at least one y such that $P(x, y)$ is false ~~for all x~~ .
12. ~~There is at least one x such that $P(x, y)$ is true for all y~~
~~There is at least one y such that $P(x, y)$ is false for all x~~ See end.
13. For any y there is at least one x such that $P(x, y, z)$ is false for at least one z .
14. ~~For at least one value of x there is no y such that $P(x, y, z)$ is false for all z or for at least one x there is more than one y such that $P(x, y, z)$ is false for all z~~
~~(obtained by rewriting the original statement as: Given any x there is one y such that $P(x, y, z)$ is true for at least one z and the y referred to is ~~a~~ ~~particular~~ unique)~~
15. The original statement rewritten is:
 Given any x there is at least one y such that $P(x, y, z)$ is true for no z or precisely one z .
 $\therefore$ the negation is:
 Given any y there is at least one x such that $P(x, y, z)$ is false for at least one z and (no z or more than one z)

Negation Exercises — Further Points Answers (3)

2

S.P. MOLYNEUX
ANALYSIS 4A

3 Oct. 1968

15. contd.

But in the above statement "at least one y " excludes the possibility of "no z ".

$\therefore$ the negation is:

Given any y there is at least one x such that $P(x, y, z)$ is false for at least one z and more than one z .

$\therefore$ The negation is:

Given any y there is at least one x such that $P(x, y, z)$ is false for more than one z .

14. For at least one x there is more than one y such that $P(x, y, z)$ is ~~true~~ ^{false} for ~~any~~ ^{any} z .
TRY AGAIN.

15. There exists an x such that for any y $P(x, y, z)$ is true for more than one z .

Question 11.

"There is at least one x s.t. $P(x, y)$ is true for all y ."

Negation: Given any x $\exists$ at least one y (possibly depending upon x) s.t. $P(x, y)$ is false.

Your negation implies that the same y serves for all x .

Consider the example: There is at least one x s.t. $(x-y)^2 > 0$ for all y . (This is false because given any x $\exists$ a $y (=x)$ s.t. $(x-y)^2 = 0$ i.e. $(x-y)^2 \not> 0$.) But it is also false that $\exists$ an x s.t. $(x-y)^2 \not> 0$ for all y & we cannot have a statement & its negation simultaneously false.

Negation Exercises — Further Points Answers (4)

Question 12.

"Given any x $\exists$ at least one y s.t. $P(x,y)$ is false"
 Negation is " $\exists$ an x s.t. $P(x,y)$ is true for all y "
 (You had it correct first time).

"There is at least one y s.t. $P(x,y)$ is false for all x "
 Negation is : "Given any y $\exists$ an x s.t. $P(x,y)$ is true."

Question 13 (A) "Given any x $\exists$ at least one y s.t. $P(x,y,z)$ is true for all z ."

Negation is (1) " $\exists$ an x s.t. for all y $\exists z$ (possibly depending upon y) s.t. $P(x,y,z)$ is false".

(2) "For any y $\exists$ at least one x s.t. $P(x,y,z)$ is false for at least one z "

Negation is (B) " $\exists$ a y s.t. $P(x,y,z)$ is true for all x and all z " Since (A) & (B) are not the same (2) cannot be the negation of (A).

[Go to Table of Contents](#)

[Go to Table of Contents](#)

7 Interpretations for Predicate Logic

- An **interpretation** is an assignment of meaning to the symbols of a **formal language**
- An interpretation often (but not always) provides a way to determine the truth values of a sentence in a formal language.
- If an interpretation assigns the value True to a sentence or theory, the interpretation is called a *model* of that sentence or theory.
- The domain is the set of all the objects being discussed.
- An interpretation assigns an object in the domain to each of the constants in the logic, and an n -ary relation on the domain to each n -ary predicate

- See Definition 12 in the Unit 6, 7 Reader

7.1 Logic Exs Interpretations Q 1

- Given the domain $\mathcal{D} = \{\text{Adam}, \text{Milton}, \text{Joan}\}$
- Consider $\forall X. ((\text{banker}(X) \wedge \text{inHedgeFund}(X)) \Rightarrow \text{sellingShort}(X))$
- In which of the following interpretations is the formula True ?

(a)

- $\mathcal{I}(\text{banker}) = \{\text{Adam}, \text{Milton}, \text{Joan}\}$
- $\mathcal{I}(\text{inHedgeFund}) = \{\text{Milton}, \text{Joan}\}$
- $\mathcal{I}(\text{sellingShort}) = \{\text{Milton}, \text{Joan}\}$

(b)

- $\mathcal{I}(\text{banker}) = \{\text{Adam}, \text{Milton}, \text{Joan}\}$
- $\mathcal{I}(\text{inHedgeFund}) = \emptyset$ ($\emptyset$ denotes the *empty set*)
- $\mathcal{I}(\text{sellingShort}) = \emptyset$

(c)

- $\mathcal{I}(\text{banker}) = \{\text{Adam}\}$
- $\mathcal{I}(\text{inHedgeFund}) = \{\text{Adam}\}$
- $\mathcal{I}(\text{sellingShort}) = \{\text{Joan}\}$

(d)

- $\mathcal{I}(\text{banker}) = \{\text{Milton}\}$
- $\mathcal{I}(\text{inHedgeFund}) = \{\text{Adam}\}$
- $\mathcal{I}(\text{sellingShort}) = \{\text{Joan}\}$

[Go to Interpretations Soln 1](#)

[Go to Table of Contents](#)

7.2 Logic Exs Interpretations Soln 1

(a) is True

(b) is True

(c) is False

(d) is True

- Give reasons for each of the above answers

[Go to Interpretations Q 1](#)

[Go to Table of Contents](#)

[Go to Table of Contents](#)

8 Logical Arguments

Validity and Justification

- There are two ways to model what counts as a logically good argument:
 - the **semantic** view
 - the **syntactic** view
- The notion of a valid argument in propositional logic is rooted in the semantic view.
- It is based on the semantic idea of interpretations: assignments of truth values to the propositional variables in the sentences under discussion.
- A *valid argument* is defined as one that preserves truth from the premises to the conclusions
- The syntactic view focuses on the syntactic form of arguments.
- Arguments which are correct according to this view are called *justified arguments*.

Proof Systems, Soundness, Completeness

- Semantic validity and syntactic justification are different ways of modelling the same intuitive property: whether an argument is logically good.
- A proof system is *sound* if any statement we can prove (justify) is also valid (true)
- A proof system is *adequate* if any valid (true) statement has a proof (justification)
- A proof system that is sound and adequate is said to be *complete*
- Propositional and predicate logic are *complete* — arguments that are valid are also justifiable and vice versa
- Unit 7 section 2.4 describes another logic where there are valid arguments that are not justifiable (provable)

Reference: [Chiswell and Hodges \(2007, page 86\)](#)

Valid arguments

- Unit 6 defines valid arguments with the notation

$$\begin{array}{c} P_1 \\ \vdots \\ P_n \\ \hline C \end{array}$$
- The argument is *valid* if and only if the value of C is *True* in each interpretation for which the value of each premise P_i is *True* for $1 \leq i \leq n$

- In some texts you see the notation $\{P_1, \dots, P_n\} \models C$
- The expression denotes a *semantic sequent* or *semantic entailment*
- The $\models$ symbol is called the *double turnstile* and is often read as *entails* or *models*
- In LaTeX $\models$ and $\models$ are produced from `\vDash` and `\models` — see also the *turnstile* package
- In Unicode $\models$ is called *TRUE* and is U+22A8, HTML `⊨`;
- The argument $\{\} \models C$ is valid if and only if C is *True* in all interpretations
- That is, if and only if C is a tautology
- Beware different notations that mean the same thing
 - Alternate symbol for empty set: $\emptyset \models C$
 - Null symbol for empty set: $\models C$
 - Original M269 notation with null axiom above the line:

$$\overline{C}$$

[Go to Table of Contents](#)

9 Justified Arguments and Natural Deduction

- Definition 7.1 An argument $\{P_1, P_2, \dots, P_n\} \vdash C$ is a justified argument if and only if either the argument is an instance of an axiom or it can be derived by means of an inference rule from one or more other justified arguments.
- Axioms

$$\Gamma \cup \{A\} \vdash A \text{ (axiom schema)}$$
- This can be read as: any formula A can be derived from the assumption (premise) of $\{A\}$ itself
- The $\vdash$ symbol is called the *turnstile* and is often read as *proves*, denoting *syntactic entailment*
- In LaTeX $\vdash$ is produced from `\vdash`
- In Unicode $\vdash$ is called *RIGHT TACK* and is U+22A2, HTML `⊢`;

See ([Thompson, 1991](#), Chp 1)

Justified Arguments — Question 1

- Show that the argument $\{P \wedge Q, S, T\} \vdash P \wedge Q$ is justified, by showing that it is an instantiation of the axiom schema.

Justified Arguments — Answer 1

- Suppose that, in the axiom schema $\Gamma \cup \{A\} \vdash A$, we instantiate Γ with $\{S, T\}$ and A with $P \wedge Q$
- Then we get the axiom $\{S, T\} \cup \{P \wedge Q\} \vdash P \wedge Q$
- Since the union $\{S, T\} \cup \{P \wedge Q\}$ is equal to $\{P \wedge Q, S, T\}$ the axiom can be written $\{P \wedge Q, S, T\} \vdash P \wedge Q$
- We use the following single line to record that the argument is justified because it is an instantiation of the axiom schema:
 1. $\{P \wedge Q, S, T\} \vdash P \wedge Q$ [Axiom]
- **Discussion** We could equally well have instantiated Γ with $\{S, T, P \wedge Q\}$ since $\{S, T, P \wedge Q\} \cup \{P \wedge Q\}$ is equal to $\{P \wedge Q, S, T\}$
- That is, a union does not produce duplicate elements.
- Notice that we begin the instantiation with a straightforward textual substitution, then simplify an expression involving sets and set operators.
- Section 2.3 of Unit 7 (not the Unit 6, 7 Reader) gives the inference rules for $\rightarrow$, $\wedge$, and $\vee$ — only dealing with *positive propositional logic* so not making use of negation — see [List of logic systems](#)
- Usually (Classical logic) have a [functionally complete](#) set of logical connectives — that is, every binary Boolean function can be expressed in terms the functions in the set

Inference Rules — Notation

- Inference rule notation:

$$\frac{Argument_1 \quad \dots \quad Argument_n}{Argument} \text{ (label)}$$

Inference Rules — Conjunction

- $\frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B}$ ($\wedge$ -introduction)
- $\frac{\Gamma \vdash A \wedge B}{\Gamma \vdash A}$ ($\wedge$ -elimination left)
- $\frac{\Gamma \vdash A \wedge B}{\Gamma \vdash B}$ ($\wedge$ -elimination right)

Inference Rules — Conjunction — Example

- Show that the argument $\{P, Q\} \vdash P \wedge Q$ is justified.
- **Answer**
 1. $\{P, Q\} \vdash P$ [Axiom]
 2. $\{P, Q\} \vdash Q$ [Axiom]
 3. $\{P, Q\} \vdash P \wedge Q$ [1,2, $\wedge$ -I]

- **Discussion** Each line consists of a number, an argument, and a justification. The axiom schema is the justification for line 1 and line 2, while line 3 is justified by applying $\wedge$ -introduction to lines 1 and 2
- The order is 1 then 2 rather than 2 then 1, corresponding to reading the first line of the rule from left to right.
- The lines above are called a *proof* of the argument $\{P, Q\} \vdash P \wedge Q$
- They are a step-by-step trace of how the argument in the final line is justified.

Justified Arguments — Question 2

- Give a proof of the argument $\{P, Q, R \vee S\} \vdash P \wedge Q$

Justified Arguments — Answer 2

1. $\{P, Q, R \vee S\} \vdash P$ [Axiom]
2. $\{P, Q, R \vee S\} \vdash Q$ [Axiom]
3. $\{P, Q, R \vee S\} \vdash P \wedge Q$ [1, 2, $\wedge$ -I]

- **Discussion** There was no need need to write down an axiom deriving the premise $R \vee S$, because we only needed the premises P and Q in order to derive $P \wedge Q$
- It would not have been wrong to begin by deriving each of the three premises in turn, though, as in the following lines:
 1. $\{P, Q, R \vee S\} \vdash P$ [Axiom]
 2. $\{P, Q, R \vee S\} \vdash Q$ [Axiom]
 3. $\{P, Q, R \vee S\} \vdash R \vee S$ [Axiom]
 4. $\{P, Q, R \vee S\} \vdash P \wedge Q$ [1, 2, $\wedge$ -I]
- One possible strategy for constructing proofs is to begin by writing down an axiom for each premise, since this gives us a way of getting started: we can always remove any unnecessary lines later.
- Of course, this might involve revising the line numbers and references to line numbers. (there are packages in LaTeX that automate this)

Justified Arguments — Question 3

- Complete the following proof to justify $\{P \wedge Q\} \vdash Q \wedge P$
 1. $\{P \wedge Q\} \vdash P \wedge Q$ [Axiom]
 2. $\{P \wedge Q\} \vdash P$ [1, $\wedge$ -E Left]
 3. $\{P \wedge Q\} \vdash Q$ [??]
 4. $\{P, Q\} \vdash Q \wedge P$ [??]

Justified Arguments — Answer 3

- Complete the following proof to justify $\{P \wedge Q\} \vdash Q \wedge P$

1. $\{P \wedge Q\} \vdash P \wedge Q$ [Axiom]
2. $\{P \wedge Q\} \vdash P$ [1, $\wedge$ -E Left]
3. $\{P \wedge Q\} \vdash Q$ [1, $\wedge$ -E Right]
4. $\{P, Q\} \vdash Q \wedge P$ [3, 2, $\wedge$ -I]

Inference Rules — Implication

- $\frac{\Gamma \cup \{A\} \vdash B}{\Gamma \vdash A \rightarrow B}$ ($\rightarrow$ -introduction)
- The above should be read as: If there is a proof (justification, inference) for **B** under the set of premises, Γ , augmented with **A**, then we have a proof (justification, inference) of **A** $\rightarrow$ **B**, *under the unaugmented set of premises, Γ* .

The unaugmented set of premises, Γ may have contained **A** already so we cannot assume

$$(\Gamma \cup \{A\}) - \{A\} \text{ is equal to } \Gamma$$

- $\frac{\Gamma \vdash A \quad \Gamma \vdash A \rightarrow B}{\Gamma \vdash B}$ ($\rightarrow$ -elimination)

Justified Arguments — Question 4

- Complete the following incomplete proof that the argument $\{P \wedge (P \rightarrow Q)\} \vdash Q$ is justified
1. $\{P \wedge (P \rightarrow Q)\} \vdash P \wedge (P \rightarrow Q)$ [??]
 2. $\{P \wedge (P \rightarrow Q)\} \vdash P$ [1, $\wedge$ -E Left]
 3. $\{P \wedge (P \rightarrow Q)\} \vdash P \rightarrow Q$ [1, ??]
 4. $\{P \wedge (P \rightarrow Q)\} \vdash Q$ [??]

Justified Arguments — Answer 4

- Complete the following incomplete proof that the argument $\{P \wedge (P \rightarrow Q)\} \vdash Q$ is justified
1. $\{P \wedge (P \rightarrow Q)\} \vdash P \wedge (P \rightarrow Q)$ [Axiom]
 2. $\{P \wedge (P \rightarrow Q)\} \vdash P$ [1, $\wedge$ -E left]
 3. $\{P \wedge (P \rightarrow Q)\} \vdash P \rightarrow Q$ [1, $\wedge$ -E right]
 4. $\{P \wedge (P \rightarrow Q)\} \vdash Q$ [2, 3, $\rightarrow$ -E]

Justified Arguments — Question 5

- Complete the following incomplete proof that the argument $\{(P \wedge Q) \rightarrow R\} \vdash P \rightarrow (Q \rightarrow R)$ is justified
1. $\{P, Q, (P \wedge Q) \rightarrow R\} \vdash P$ [Axiom]
 2. $\{P, Q, (P \wedge Q) \rightarrow R\} \vdash Q$ [??]
 3. $\{P, Q, (P \wedge Q) \rightarrow R\} \vdash (P \wedge Q) \rightarrow R$ [Axiom]
 4. $\{P, Q, (P \wedge Q) \rightarrow R\} \vdash P \wedge Q$ [??]
 5. $\{P, Q, (P \wedge Q) \rightarrow R\} \vdash R$ [4, 3, $\rightarrow$ -E]
 6. $\{P, (P \wedge Q) \rightarrow R\} \vdash Q \rightarrow R$ [5, $\rightarrow$ -I]
 7. $\{(P \wedge Q) \rightarrow R\} \vdash P \rightarrow (Q \rightarrow R)$ [6, ??]

Justified Arguments — Answer 5

- Complete the following incomplete proof that the argument $\{(P \wedge Q) \rightarrow R\} \vdash P \rightarrow (Q \rightarrow R)$ is justified
 1. $\{P, Q, (P \wedge Q) \rightarrow R\} \vdash P$ [Axiom]
 2. $\{P, Q, (P \wedge Q) \rightarrow R\} \vdash Q$ [Axiom]
 3. $\{P, Q, (P \wedge Q) \rightarrow R\} \vdash (P \wedge Q) \rightarrow R$ [Axiom]
 4. $\{P, Q, (P \wedge Q) \rightarrow R\} \vdash P \wedge Q$ [1, 2, $\wedge$ -I]
 5. $\{P, Q, (P \wedge Q) \rightarrow R\} \vdash R$ [4, 3, $\rightarrow$ -E]
 6. $\{P, (P \wedge Q) \rightarrow R\} \vdash Q \rightarrow R$ [5, $\rightarrow$ -I]
 7. $\{(P \wedge Q) \rightarrow R\} \vdash P \rightarrow (Q \rightarrow R)$ [6, $\rightarrow$ -I]

Inference Rules — Disjunction

- $\frac{\Gamma \vdash A}{\Gamma \vdash A \vee B}$ ($\vee$ -introduction left)
- $\frac{\Gamma \vdash B}{\Gamma \vdash A \vee B}$ ($\vee$ -introduction right)
- Disjunction elimination

$$\frac{\Gamma \vdash A \vee B \quad \Gamma \cup \{A\} \vdash C \quad \Gamma \cup \{B\} \vdash C}{\Gamma \vdash C} \text{ ($\vee$ -elimination)}$$

- The above should be read: if a set of premises Γ justifies the conclusion $A \vee B$ and Γ augmented with each of A or B separately justifies C , then Γ justifies C
- Disjunction elimination is a formal version of *proof by case analysis*

Reference: [O'Donnell et al. \(2006, page 137\)](#)

Disjunction — Example 1

- Show that the argument $\{P\} \vdash P \vee Q$ is justified.
- **Answer**
 1. $\{P\} \vdash P$ [Axiom]
 2. $\{P\} \vdash P \vee Q$ [1, $\vee$ -I left]

Justified Arguments — Question 6

- Show that the argument $\{Q\} \vdash P \vee Q$ is justified

Justified Arguments — Answer 6

1. $\{Q\} \vdash Q$ [Axiom]
2. $\{Q\} \vdash P \vee Q$ [1, $\vee$ -I right]

Disjunction — Example 2

- Show that the argument $\{P \vee Q\} \vdash Q \vee P$ is justified.

- **Answer**

1. $\{P \vee Q\} \vdash P \vee Q$ [Axiom]
2. $\{P \vee Q, P\} \vdash P$ [Axiom]
3. $\{P \vee Q, P\} \vdash Q \vee P$ [2, $\vee$ -I right]
4. $\{P \vee Q, Q\} \vdash Q$ [Axiom]
5. $\{P \vee Q, Q\} \vdash Q \vee P$ [4, $\vee$ -I left]
6. $\{P \vee Q\} \vdash Q \vee P$ [1, 3, 5, $\vee$ -E]

Justified Arguments — Question 7

- Complete the following incomplete proof that the argument $\{Q \rightarrow R\} \vdash (P \vee Q) \rightarrow (P \vee R)$ is justified

1. $\{Q \rightarrow R, P \vee Q, Q\} \vdash Q \rightarrow R$ [Axiom]
2. $\{Q \rightarrow R, P \vee Q\} \vdash P \vee Q$ [??]
3. $\{Q \rightarrow R, P \vee Q, P\} \vdash P$ [??]
4. $\{Q \rightarrow R, P \vee Q, P\} \vdash P \vee R$ [?? $\vee$ -I left]
5. $\{Q \rightarrow R, P \vee Q, Q\} \vdash Q$ [Axiom]
6. $\{Q \rightarrow R, P \vee Q, Q\} \vdash R$ [5, 1, $\rightarrow$ -E]
7. $\{Q \rightarrow R, P \vee Q, Q\} \vdash P \vee R$ [6, ??]
8. $\{Q \rightarrow R, P \vee Q\} \vdash P \vee R$ [2, 4, 7, $\vee$ -E]
9. $\{Q \rightarrow R\} \vdash (P \vee Q) \rightarrow (P \vee R)$ [?? $\rightarrow$ -I]

Justified Arguments — Answer 7

- Complete the following incomplete proof that the argument $\{Q \rightarrow R\} \vdash (P \vee Q) \rightarrow (P \vee R)$ is justified

1. $\{Q \rightarrow R, P \vee Q, Q\} \vdash Q \rightarrow R$ [Axiom]
2. $\{Q \rightarrow R, P \vee Q\} \vdash P \vee Q$ [Axiom]
3. $\{Q \rightarrow R, P \vee Q, P\} \vdash P$ [Axiom]
4. $\{Q \rightarrow R, P \vee Q, P\} \vdash P \vee R$ [3, $\vee$ -I left]
5. $\{Q \rightarrow R, P \vee Q, Q\} \vdash Q$ [Axiom]
6. $\{Q \rightarrow R, P \vee Q, Q\} \vdash R$ [5, 1, $\rightarrow$ -E]
7. $\{Q \rightarrow R, P \vee Q, Q\} \vdash P \vee R$ [6, $\vee$ -I right]
8. $\{Q \rightarrow R, P \vee Q\} \vdash P \vee R$ [2, 4, 7, $\vee$ -E]
9. $\{Q \rightarrow R\} \vdash (P \vee Q) \rightarrow (P \vee R)$ [8, $\rightarrow$ -I]

9.1 Proofs in Tree Form

- The syntax of proofs is recursive:
- A proof is either an axiom, or the result of applying a rule of inference to one, two or three proofs.
- We can therefore represent a proof by a tree diagram in which each node have one, two or three children

- For example, the proof of $\{P \wedge (P \rightarrow Q)\} \vdash Q$ in *Question 4* can be represented by the following diagram:

$$\frac{\frac{\{P \wedge (P \rightarrow Q)\} \vdash P \wedge (P \rightarrow Q)}{\{P \wedge (P \rightarrow Q)\} \vdash P} (\wedge\text{-E left}) \quad \frac{\{P \wedge (P \rightarrow Q)\} \vdash P \wedge (P \rightarrow Q)}{\{P \wedge (P \rightarrow Q)\} \vdash P \rightarrow Q} (\wedge\text{-E right})}{\{P \wedge (P \rightarrow Q)\} \vdash Q} (\rightarrow\text{-E})$$

Justified Arguments — Question 8

- Draw a diagram to represent the following proof:

1. $\{P, R\} \vdash P$ [Axiom]
2. $\{P, R\} \vdash R$ [Axiom]
2. $\{P, R\} \vdash P \wedge R$ [1, 2, $\wedge$ -I]

Justified Arguments — Answer 8

$$\frac{\{P, R\} \vdash P \quad \{P, R\} \vdash R}{\{P, R\} \vdash P \wedge R} (\wedge\text{-I})$$

[Go to Table of Contents](#)

9.2 Self-Assessment activity 7.4

- Is the following a justified argument ?
- $\{P \rightarrow R, Q \rightarrow R, P \vee Q\} \vdash R$
- First of all, prove
 - $\{P \rightarrow R, Q \rightarrow R, P \vee Q\} \vdash P \vee Q$
 - $\{P \rightarrow R, Q \rightarrow R, P \vee Q\} \cup \{P\} \vdash R$
 - $\{P \rightarrow R, Q \rightarrow R, P \vee Q\} \cup \{Q\} \vdash R$

Self-Assessment activity 7.4 — Tree layout

- Let $\Gamma = \{P \rightarrow R, Q \rightarrow R, P \vee Q\}$
- $$\frac{\Gamma \vdash P \vee Q \quad \Gamma \cup \{P\} \vdash R \quad \Gamma \cup \{Q\} \vdash R}{\Gamma \vdash R} (\vee\text{-elimination})$$
- $$\frac{\Gamma \cup \{P\} \vdash P \quad \Gamma \cup \{P\} \vdash P \rightarrow R}{\Gamma \cup \{P\} \vdash R} (\rightarrow\text{-elimination})$$
- $$\frac{\Gamma \cup \{Q\} \vdash Q \quad \Gamma \cup \{Q\} \vdash Q \rightarrow R}{\Gamma \cup \{Q\} \vdash R} (\rightarrow\text{-elimination})$$
- Complete tree layout
- $$\frac{\Gamma \vdash P \vee Q \quad \frac{\Gamma \cup \{P\} \vdash P \quad \Gamma \cup \{P\} \vdash P \rightarrow R}{\Gamma \cup \{P\} \vdash R} (\rightarrow\text{-E}) \quad \frac{\Gamma \cup \{Q\} \vdash Q \quad \Gamma \cup \{Q\} \vdash Q \rightarrow R}{\Gamma \cup \{Q\} \vdash R} (\rightarrow\text{-E})}{\Gamma \vdash R} (\vee\text{-E})$$

Self-assessment activity 7.4 — Linear Layout

1. $\{P \rightarrow R, Q \rightarrow R, P \vee Q\} \vdash P \vee Q$ [Axiom]
2. $\{P \rightarrow R, Q \rightarrow R, P \vee Q\} \cup \{P\} \vdash P$ [Axiom]
3. $\{P \rightarrow R, Q \rightarrow R, P \vee Q\} \cup \{P\} \vdash P \rightarrow R$ [Axiom]
4. $\{P \rightarrow R, Q \rightarrow R, P \vee Q\} \cup \{Q\} \vdash Q$ [Axiom]
5. $\{P \rightarrow R, Q \rightarrow R, P \vee Q\} \cup \{Q\} \vdash Q \rightarrow R$ [Axiom]
6. $\{P \rightarrow R, Q \rightarrow R, P \vee Q\} \cup \{P\} \vdash R$ [2, 3, $\rightarrow$ -E]
7. $\{P \rightarrow R, Q \rightarrow R, P \vee Q\} \cup \{Q\} \vdash R$ [4, 5, $\rightarrow$ -E]
8. $\{P \rightarrow R, Q \rightarrow R, P \vee Q\} \vdash R$ [1, 6, 7, $\vee$ -E]

[Go to Table of Contents](#)

[Go to Table of Contents](#)

10 Calculating with Logic

10.1 Logic Puzzles — Introduction

- The following puzzles are usually given as exercises in verbal reasoning — however you can use your knowledge of propositional logic to *calculate* the answers.
- The answers below (in the notes version) give references to the sources of the puzzles and solutions.

[Go to Table of Contents](#)

10.2 Knights and Knaves

- There is a wide variety of puzzles about an island in which certain inhabitants called *knights* always tell the truth, and others called *knaves* always lie.
 - It is assumed that every inhabitant of the island is either a knight or a knave.
 - The following puzzles can be solved by verbal reasoning or by using truth tables
1. Three inhabitants of this island — A, B and C — are standing together in a garden. You pass by and ask A *Are you a knight or a knave?* A answers but rather indistinctly so you cannot hear. You then ask B *What did A say?* B replies *A said that he is a knave* At this point C says *Don't believe B; he is lying*
What are B and C?
 2. Suppose instead of asking A what he is, you asked A *How many knights are among you?* Again you cannot hear A's reply. So you ask B *What did A say?* B replies *A said there is only one knight among us* Then C says *Don't believe B; he is lying*
Now what are B and C?
 3. In this problem there are only two people A and B each of whom is either a knight or knave. A makes the following statement *At least one of us is a knave*
What are A and B?

4. Suppose A says *Either I am a knave or B is a knight*

What are A and B ?

5. Suppose A says *Either I am a knave or else $2 + 2 = 5$*

What would you conclude ?

6. Again we have 3 people A B C each either a knave or a knight. A and B say the following:

A: All of us are knaves

B: Exactly one of us is a knight

What are A B C ?

7. Again three inhabitants A B C each of whom is either a knight or knave. Two people are said to be of the same type if they are both knights or both knaves. A and B make the following statements:

A: B is a knave

B: A and C are of the same type

What is C ?

8. Again three people A B C. A says *B and C are of the same type* Someone then asks C *Are A and B of the same type ?*

What does C answer ?

[Go to Table of Contents](#)

10.3 Knights and Knaves — Variant

- A variation on the above type of problems deals with three types of people: knights and knaves as before and normal people who sometimes lie and sometimes tell the truth.

9. We are given three people A B C one of whom is a knight, one a knave and one normal (but maybe not in that order). They make the following statements:

A: I am normal

B: That is true

C: Exactly one of us is a knave

What are A B C ?

10. Two people A and B each of whom is either a knight, or knave or normal make the following statements:

A: B is a knight

B: A is not a knight

Prove that at least one of them is telling the truth but is not a knight.

11. This time A and B say the following:

A: B is a knight

B: A is a knave

Prove that either one of them is telling the truth but is not a knight or one of them is lying but is not a knave.

[Go to Table of Contents](#)

10.4 Harder Logic Puzzles

- Here are several logic puzzles which involve liars, truth-tellers and those who speak the truth or lie at random.
- The later puzzles are actually extensions of the first (so if you have really solved the first, the rest might be easier).

1.

- A tourist is enjoying an afternoon refreshment in a local pub in England when the bartender says to him: “Do you see those three men over there ? One is Mr. X, who always tells the truth, another is Mr. Y, who always lies, and the third is Mr. Z, who sometimes tells the truth and sometimes lies (that is, Mr. Z answers *yes* or *no* at random without regard for the question). You may ask them three yes/no questions, always indicating which man should answer. If, after asking these three questions, you correctly identify Mr. X, Mr. Y, and Mr. Z, they will buy you a drink.”
- What yes/no questions should the thirsty tourist ask ?
- Hint: Use the first question to find some person of the three who is not Mr. Z. Ask him the other two questions.

2.

- In a certain country, there are three kinds of people: workers (who always tell the truth), capitalists (who never tell the truth), and students (who sometimes tell the truth and sometimes lie).
- At a fork in the road, one branch leads to the capital. A worker, a capitalist, and a student are standing at the side of the road but are not identifiable in any obvious way.
- By asking two yes or no questions, find out which fork leads to the capital. (Each question may be addressed to any of the three.)

3.

- Three gods A, B, and C are called, in some order, True, False, and Random. True always speaks truly, False always speaks falsely, but whether Random speaks truly or falsely is a completely *random* matter.
- Your task is to determine the identities of A, B, and C by asking three questions; each question must be put to exactly one god.
- The gods understand English, but will answer all questions in their own language, in which the words for “yes” and “no” are “da” and “ja”, in some order. *You do not know which word means which.*

[Go to Table of Contents](#)

10.5 Knights and Knaves — Answers

This set of exercises (and the answers) all come from [Smullyan \(1981\)](#). Without assuming *any* prior knowledge of logic, Smullyan takes the reader through a logic course using verbal reasoning. More conventional books on logic and computing (such as [Manna \(1974\)](#), [Mendelson \(1987\)](#), [Backhouse \(1986\)](#), [Backhouse \(2003\)](#)) use some of these puzzles as part of their sections on logic. Other puzzle books by Smullyan include an introduction to Gödel's *Incompleteness Theorem*, combinatory logic and Turing machines — see the references at the end.

1. Problem 26 in [Smullyan \(1981\)](#)

It is impossible for either a knight or a knave to say, *I'm a knave*, because a knight wouldn't make the false statement that he is a knave, and a knave wouldn't make the true statement that he is a knave. Therefore A never did say that he was a knave. So B lied when he said that A said that he was a knave. Hence B is a knave. Since C said that B was lying and B was indeed lying, then C spoke the truth, hence he is a knight. Thus B is a knave and C is a knight. (It is impossible to know what A is.)

2. Problem 27 in [Smullyan \(1981\)](#)

The answer is the same as that of the preceding problem, though the reasoning is a bit different.

The first thing to observe is that B and C must be of opposite types, since B contradicts C. So of these two, one is a knight and the other a knave. Now, if A were a knight, then there would be two knights present, hence A would not have lied and said there was only one. On the other hand, if A were a knave, then it would be true that there was exactly one knight present; but then A, being a knave, couldn't have made that true statement. Therefore A could not have said that there was one knight among them. So B falsely reported A's statement, and thus B is a knave and C is a knight.

3. Problem 28 in [Smullyan \(1981\)](#)

Suppose A were a knave. Then the statement *At least one of us is a knave* would be false (since knaves make false statements); hence they would be both knights. Thus if A were a knave he would also have to be a knight, which is impossible. Therefore A is not a knave; he is a knight. Therefore his statement must be true, so at least one of them really is a knave. Since A is a knight, then B must be the knave. So A is a knight and B is a knave.

4. Problem 29 in [Smullyan \(1981\)](#)

This problem is a good introduction to the logic of disjunction. Given any two statements P, Q, the statement *either P or Q* means that at least one (and possibly both) of the statements P, Q are true. If the statement *either P or Q* should be false, then both the statements P, Q are false. For example, if I should say, *Either it is raining or it is snowing*, then if my statement is incorrect, it is both false that it is raining and false that it is snowing.

This is the way *either/or* is used in logic. . . In daily life it is sometimes used this way (allowing the possibility that both alternatives hold) and sometimes in the *exclusive* sense — that one and only one of the conditions holds. As an example of the exclusive use, if I say, *I will marry Betty or I will marry Jane*, it is understood that the two possibilities are mutually exclusive — that is I will not marry both girls. On the other hand if a college catalogue states that an entering student is required to have had either a year of mathematics or a year of a foreign language, the college is certainly not going to exclude you if you had both ! This is the *inclusive* use of *either/or* and is the one we will constantly employ.

Another important property of the disjunction relation *either this or that* is this. Consider the statement *P or Q* (which is short for *either P or Q*). Suppose the statement happens to be true. Then if P is false, Q must be true (because at least one of them is true, so if P is false, Q must be the true one). For example, suppose it is true that it is either raining or snowing, but it is false that it is raining. Then it must be true that it is snowing.

We apply these two principles as follows. A made a statement of the disjunctive type: *Either I am a knave or B is a knight*. Suppose A to be a knave. Then the above statement must be false. This means that it is neither true that A is a knave nor that B is a knight. So if A were a knave then it would follow that he is not a knave — which would be a contradiction. Therefore A must be a knight.

We have thus established that A is a knight. Therefore his statement is true that at least one of the possibilities holds: (1) A is a knave; (2) B is a knight. Since possibility (1) is false (since A is a knight) then possibility (2) must be the correct one, i.e. B is a knight. Hence A, B are both knights.

5. Problem 30 in Smullyan (1981)

The only valid conclusion is that the author of this problem is not a knight. The fact is that neither a knight nor a knave could possibly make such a statement. If A were a knight then the statement that either A is a knave or that $2 + 2 = 5$ would be false, since it is neither the case that A is a knave nor that $2 + 2 = 5$. Thus A, a knight, would have made a false statement, which is impossible. On the other hand, if A were a knave then the statement that either A is a knave or that $2 + 2 = 5$ would be true, since the first clause that A is a knave is true. Thus A, a knave, would have made a true statement, which is equally impossible.

Therefore the conditions of the problem are contradictory. (*Sneaky !*)

6. Problem 31 in Smullyan (1981)

To begin with, A must be a knave, for if he were a knight, then it would be true that all three are knaves and hence that A too is a knave. If A were a knight he would have to be a knave, which is impossible. So A is a knave. Hence his statement was false, so in fact there is at least one knight among them.

Now, suppose B were a knave. Then A and B would both be knaves, so C would be a knight (since there is at least one knight among them). This would mean that there was exactly one knight among them, hence B's statement would be true. We would thus have the impossibility of a knave making a true statement. Therefore B must be a knight.

We now know that A is a knave and that B is a knight. Since B is a knight, his statement is true, so there is exactly one knight among them. This knight must be B, hence C

must be a knave.

7. Problem 34 in Smullyan (1981)

Suppose A is a knight. Then his statement that B is a knave must be true, so B is then a knave. Hence B's statement that A and C are of the same type is false, so A and C are of different types. Hence C must be a knave (since A is a knight). Thus if A is a knight, then C is a knave.

On the other hand, suppose A is a knave. Then his statement that B is a knave is false, hence B is a knight. Hence B's statement is true that A and C are of the same type. This means that C must be a knave (since A is)

We have shown that regardless of whether A is a knight or a knave, C must be a knave. Hence C is a knave.

8. Problem 35 in Smullyan (1981)

This is solved by an analysis into cases.

CASE ONE: A is a knight. Then B, C really are of the same type. If C is a knight, then B is also a knight, hence is of the same type as A, so C being truthful must answer yes. If C is a knave, then B is also a knave (since he is of the same type as C), hence is of different type to A. So C, being a knave, must lie and say yes.

CASE TWO: A is a knave. Then B, C are of different types. If C is a knight, then B is a knave, hence he is of the same type as A. So C, being a knight must answer yes. If C is a knave, then B, being of different type than C, is a knight, hence is of different type than A. Then C, being a knave, must lie about A and B being of different types, so he will answer yes.

Thus in both cases C answers yes

A variation on the above type of problems deals with three types of people: knights and knaves as before and normal people who sometimes lie and sometimes tell the truth.

9. Problem 39 in Smullyan (1981)

To begin with, A cannot be a knight, because a knight would never say that he is normal. So A is a knave or is normal. Suppose A were normal. Then B's statement would be true, hence B is a knight or a normal, but B can't be normal (since A is), so B is a knight. This leaves C a knave. But a knave cannot say he is not normal (because a knave really isn't normal), so we have a contradiction. Therefore A cannot be normal. Hence A is a knave. Then B's statement is false, so B must be normal (he can't be a knave since A is). Thus A is the knave, B is the normal, hence C is the knight.

10. Problem 40 in Smullyan (1981)

The interesting thing about this problem is that it is impossible to know whether it is A who is telling the truth but isn't a knight or whether it is B who is telling the truth but isn't a knight; all we can prove is that at least one of them has that property.

Either A is telling the truth or he isn't. We shall prove: (1) If he is, then A is telling the truth but isn't a knight; (2) If he isn't, then B is telling the truth but isn't a knight.

(1) Suppose A is telling the truth. Then B really is a knight. Hence B is telling the truth, so A isn't a knight. Thus if A is telling the truth then A is a person who is telling the truth but isn't a knight.

(2) Suppose A is not telling the truth. Then B isn't a knight. But B must be telling the truth, since A can't be a knight (because A is not telling the truth). So in this case B is telling the truth but isn't a knight.

11. Problem 41 in [Smullyan \(1981\)](#)

We shall show that if B is telling the truth then he isn't a knight, and if he isn't telling the truth then A is lying but isn't a knave.

(1) Suppose B is telling the truth. Then A is a knave, hence A is certainly not telling the truth, hence B is not a knight. So in this case B is telling the truth but isn't a knight.

(2) Suppose B is not telling the truth. Then A is not really a knave. But A is certainly lying about B, because B cannot be a knight if he isn't telling the truth. So in this case A is lying but isn't a knave.

[Go to Table of Contents](#)

10.6 Knights and Knaves — Truth Table Answers

This set of exercises (and the answers) all come from [Smullyan \(1981\)](#). Without assuming *any* prior knowledge of logic, Smullyan takes the reader through a logic course using verbal reasoning. More conventional books on logic and computing (such as [Manna \(1974\)](#), [Mendelson \(1987\)](#), [Backhouse \(1986\)](#), [Backhouse \(2003\)](#)) use some of these puzzles as part of their sections on logic.

The set of solutions below use the truth table approach outlined in [Backhouse \(1986\)](#) — though any good book with a section on logic would contain details (see, for example, [Devlin \(1992, page 13\)](#))

- We will use the following notation
 - A stands for *A is a Knight*
 - **not**A stands for *A is a Knave*
 - SA stands for *The statement by A is **True***
 - **not**SA stands for *The statement by A is **False***
- Hence, in this world of truth tellers and liars we know:
 - $(A \Rightarrow SA) \text{ and } (\text{not } A \Rightarrow \text{not } SA)$
- The above is equivalent to:
 - $A \iff SA$
- This gives us a way of solving the puzzles using truth tables

Q 1 (Problem 26 in [Smullyan \(1981\)](#))

- We have the following from the statements of B and C:
 - $B \iff (A \iff \text{not } A)$
 - $C \iff \text{not } B$

- We now construct a truth table for the conjunction of the two propositions and see which entries are **True**.

B	C	$B \iff (A \iff \text{not } A)$		and	$C \iff \text{not } B$
True	True	False	False	False	False
True	False	False	False	False	True
False	True	True	False	True	True
False	False	True	False	False	False

- The **True** tells us that B is a knave and C is a knight — it is the only entry in the truth table for the proposition which is **True**.

Q 2 (Problem 27 in Smullyan (1981))

- The answer is the same as that of the preceding problem, though the reasoning is a bit different.
- We have the following from the statements of B and C:
 - $B \iff (A \iff 1 \text{ knight})$
 - $C \iff \text{not } B$

A	B	C	$B \iff (A \iff 1 \text{ knight})$		and	$C \iff \text{not } B$
True	True	True	False	False	False	False
True	True	False	False	False	False	True
True	False	True	True	False	True	True
True	False	False	False	True	False	False
False	True	True	True	True	False	False
False	True	False	False	False	False	True
False	False	True	True	False	True	True
False	False	False	False	True	False	False

Q 3 (Problem 28 in Smullyan (1981))

- We have the following from the statement of A:
 - $A \iff 1 \text{ or more knaves}$

A	B	$A \iff 1 \text{ or more knaves}$	
True	True	False	False
True	False	True	True
False	True	False	True
False	False	False	True

- So A is a knight and B is a knave.

Q 4 (Problem 29 in Smullyan (1981))

- We have the following from the statement of A:
 - $A \iff (\text{not } A \text{ or } B)$

A	B	$A \iff (\text{not } A \text{ or } B)$	
True	True	True	True
True	False	False	False
False	True	False	True
False	False	False	True

- So we have A and B are both knights.

Q 5 (Problem 30 in Smullyan (1981))

- We have the following from the statement of A:

$$- A \iff (\text{not } A \text{ or } 2 + 2 = 5)$$

A	$A \iff (\text{not } A \text{ or } 2 + 2 = 5)$	
True	False	False
False	False	True

- So here there is no solution for any possible assignment of truth values — we call this a contradiction.

Q 6 (Problem 31 in Smullyan (1981))

- We have the following statements:

- SA: All Knaves
- SB: Exactly 1 knave

A	B	C	$(A \iff SA)$	and	$(B \iff SB)$
True	True	True	False	False	False
True	True	False	False	False	True
True	False	True	False	False	False
True	False	False	False	False	True
False	True	True	True	True	True
False	True	False	True	False	False
False	False	True	True	False	False
False	False	False	False	False	True

- So we have A knave and B and C knights.

Q 7 (Problem 34 in Smullyan (1981))

- We have the following statements:

- SA: not B
- SB: A & C are the same

A	B	C	$(A \iff SA)$	and	$(B \iff SB)$
True	True	True	False	False	True
True	True	False	False	False	False
True	False	True	True	False	False
True	False	False	True	True	True
False	True	True	False	False	False
False	True	False	True	True	True
False	False	True	False	False	True
False	False	False	True	False	False

- So C must be a knave.

Q 8 (Problem 35 in [Smullyan \(1981\)](#))

- We have the following statement:
 - SA: B & C are same
- We now construct a truth table including the response of C to the question *Is it **True** that A and B are the same?*

A	B	C	$(A \iff SA)$	and	$(C \iff A = B)$
True	True	True	True	True	True yes
True	True	False	False	False	False no
True	False	True	False	False	False no
True	False	False	True	True	True yes
False	True	True	False	False	False no
False	True	False	True	True	True yes
False	False	True	True	True	True yes
False	False	False	False	False	False no

- Thus in both cases C answers *yes*

[Go to Table of Contents](#)

10.7 Harder Logic Puzzles — Solutions

- The approach to finding a solution is based on the answer to exercise 1.22 in [Backhouse \(1986\)](#), pages 43,233)
- The source of the problems is as follows:
 - Q 1 is Problem 2-7(b) in [Manna \(1974\)](#)
 - Q 2 is exercise 1.46(b) in [Mendelson \(1997\)](#), page 24)
 - Q 3 is from chapter 29 of [Boolos \(1998\)](#)

This problem was originally in an article by George Boolos in *The Harvard Review of Philosophy* 6 (1996): 62-65

Q 1 (Problem 2-7(b) in Manna (1974))

- A tourist is enjoying an afternoon refreshment in a local pub in England when the bartender says to him: “Do you see those three men over there ? One is Mr. X, who always tells the truth, another is Mr. Y, who always lies, and the third is Mr. Z, who sometimes tells the truth and sometimes lies (that is, Mr. Z answers *yes* or *no* at random without regard for the question). You may ask them three yes/no questions, always indicating which man should answer. If, after asking these three questions, you correctly identify Mr. X, Mr. Y, and Mr. Z, they will buy you a drink.”
- What yes/no questions should the thirsty tourist ask ?
- Hint: Use the first question to find some person of the three who is not Mr. Z. Ask him the other two questions.

Q 1 Solution (a)

- We can label the people (say by distance from us) as A, B and C.
- With no prior knowledge we may as well ask the first question to A.
- A could be a knight, a knave or a normal (that’s what we call people who lie or tell the truth at random).
- The hint tells us that if should use the first question to identify someone who is not normal.
- Once we have done that the rest is easy: ask a knight or a knave if $2 + 2 = 5$ and you immediately know what they are and can then use them to tell you who the rest are with one question.

Q 1 Solution (b)

- The *Eureka* step is to realise that you can *calculate* the first question by working out what properties it must have and then rearranging a description of the properties as propositions into the form:
- $Q \iff \text{some proposition not involving } Q$
- where Q stands for a question of the form “Is it **True** that ...” where the question is trying to identify whether B is normal or not.

Q 1 Solution (c)

- In this case our first question (to A) should satisfy the following:
 1. If A is a knight and A says Q is **True** then B is normal.
 2. If A is a knave and A says Q is **True** then B is normal.
 3. If A is a knight and A says Q is **False** then B is **not** normal.
 4. If A is a knave and A says Q is **False** then B is **not** normal.

Q 1 Solution (d)

- We can represent the above statements as a compound proposition.
- We use “BN” to represent “B is normal”; “Q” stands for “Q is **True**”.
- Remember that if a knave says “Q is **True**” that “**not** Q” is really the case (and vice-versa).

$(A \text{ and } Q) \Rightarrow \text{BN}$

and

$(\text{not } A \text{ and not } Q) \Rightarrow \text{BN}$

and

$(A \text{ and not } Q) \Rightarrow \text{not BN}$

and

$(\text{not } A \text{ and } Q) \Rightarrow \text{not BN}$

Q 1 Solution (e)

- We now use the following identity (use a truth table to prove the identity):
- $(p \text{ and } q) \Rightarrow r \equiv q \Rightarrow (p \Rightarrow r)$
- This gives us:

$Q \Rightarrow (A \Rightarrow \text{BN})$

and

$\text{not } Q \Rightarrow (\text{not } A \Rightarrow \text{BN})$

and

$\text{not } Q \Rightarrow (A \Rightarrow \text{not BN})$

and

$Q \Rightarrow (\text{not } A \Rightarrow \text{not BN})$

Q 1 Solution (f)

- We now use the following identity (again prove that this is an identity):
- $(p \Rightarrow q) \text{ and } (p \Rightarrow r) \equiv p \Rightarrow (q \text{ and } r)$
- This gives us:

$Q \Rightarrow ((A \Rightarrow \text{BN}) \text{ and } (\text{not } A \Rightarrow \text{not BN}))$

and

$\text{not } Q \Rightarrow ((\text{not } A \Rightarrow \text{BN}) \text{ and } (A \Rightarrow \text{not BN}))$

Q 1 Solution (g)

- We now use the following identity (again prove that this is an identity):
- $(\text{not } p \Rightarrow \text{not } q) \equiv p \Rightarrow q$
- This gives us:

$$Q \Rightarrow ((A \Rightarrow BN) \text{ and } (BN \Rightarrow A))$$

and

$$\text{not } Q \Rightarrow ((\text{not } A \Rightarrow BN) \text{ and } (BN \Rightarrow \text{not } A))$$

- Use the definition of $\Leftrightarrow$

$$Q \Rightarrow (A \Leftrightarrow BN)$$

and

$$\text{not } Q \Rightarrow (\text{not } A \Leftrightarrow BN)$$

Q 1 Solution (h)

- We finally use the definition of $\Leftrightarrow$ and the identity:
- $\text{not } p \Leftrightarrow q \equiv \text{not}(p \Leftrightarrow q)$
- This gives us:

$$Q \Leftrightarrow (A \Leftrightarrow BN)$$

- So in English our first question (to A) would be:
 - *Is it true that the statement that you are a truth teller is equivalent to the statement that B is normal ?*
- This gives a general approach to similar puzzles

Q 1 (f) to (g) steps

- From step (f) using $\text{not } p \Rightarrow \text{not } q \equiv q \Rightarrow p$

$$Q \Rightarrow ((A \Rightarrow BN) \text{ and } (BN \Rightarrow A))$$

and

$$\text{not } Q \Rightarrow ((\text{not } A \Rightarrow BN) \text{ and } (BN \Rightarrow \text{not } A))$$

- Using the definition of $\iff$

$$Q \Rightarrow (A \iff BN)$$

and

$$\text{not } Q \Rightarrow (\text{not } A \iff BN)$$

- Using $\text{not } p \iff q \equiv \text{not}(p \iff q)$

$$Q \Rightarrow (A \iff BN)$$

and

$$\text{not } Q \Rightarrow \text{not}(A \iff BN)$$

- Using $\text{not } p \Rightarrow \text{not } q \equiv q \Rightarrow p$

$$Q \Rightarrow (A \iff BN)$$

and

$$(A \iff BN) \Rightarrow Q$$

- Using the definition of $\iff$ then gives us step (g)

[Go to Table of Contents](#)

[Go to Table of Contents](#)

11 Logic and Programming

- [Curry-Howard isomorphism](#) is the direct relationship between computer programs and mathematical proofs
- A proof is a program
- The formula it proves is the type for the program
- A logic corresponds to a programming language
- For example, at the level of formulas and types:
- Implication $\leftrightarrow$ function type
- Conjunction (AND) $\leftrightarrow$ product type
- Disjunction (OR) $\leftrightarrow$ sum type
- [Haskell/The Curry-Howard isomorphism](#) — article on CH and the functional programming language Haskell
- [Curry-Howard isomorphism](#) — overview article

[Go to Table of Contents](#)

12 Future Work

- Wednesday 28 April 2021 iCMA46 due
- Sunday, 2 May 2021 online tutorial Unit 7 Computability, Complexity
- Sunday, 16 May 2021 online tutorial exam revision
- Saturday, 22 May 2021 online tutorial exam revision
- Tuesday 25 May 2021 iCMA47 due
- Tuesday 8 June 2021 Exam
- Please email me with any requests for particular topics

13 Web Sites & References

13.1 Web Sites

- [Truth function](#) — the sixteen possible truth functions of two boolean inputs
- [Logical connective](#) — some historical context
- [Functional completeness](#) — what subsets of the truth functions could you use
- [Sheffer stroke](#) — also known as [NAND](#) (not and) and *alternative denial* — [{NAND}](#) is a functionally complete set — you can construct all elements of {AND, OR, NOT} — you could do all your propositional logic just using this
- [Logical NOR](#) — also known as Pierce's arrow — also functionally complete — the [Apollo Guidance Computer](#) was implemented in [NOR](#) gates alone. See also [Engineering & Technology Magazine article on AGC, tales from the Lunar Module Guidance Computer](#)
- [List of Logic Symbols](#)

[Go to Table of Contents](#)

References

- Backhouse, R. (2003). *Program Construction: Calculating Implementations from Specifications*. Wiley. ISBN 0470848820. URL <http://www.cs.nott.ac.uk/~rcb/>. 45, 48
- Backhouse, Roland C. (1986). *Program Construction and Verification*. Prentic-Hall. ISBN 0-13-729146-9. 45, 48, 51
- Boolos, George (1998). *Logic, Logic and Logic*. Harvard University Press. ISBN 0-674-53766-1. 51
- Bornat, Richard (2005). *Proof and Disproof in Formal Logic: An Introduction for programmers*. Oxford. ISBN 0198530277.
- Carroll, Lewis and Martin Gardner (1967). *The Annotated Snark*. Penguin. Original edition 1962.

- Carroll, Lewis and Martin Gardner (2001). *The Annotated Alice: The Definitive Edition*. Penguin. ISBN 0140289291.
- Carroll, Lewis; Martin Gardner; Henry Holiday; and Adam Gopnik (2006). *The Annotated Hunting of the Snark*. W. W. Norton Company. ISBN 0393062422.
- Carroll, Lewis; Martin Gardner; and John Tenniel (1970). *The Annotated Alice : Alice's Adventures in Wonderland and Through the Looking Glass*. Penguin Books. ISBN 0140013873.
- Chiswell, Ian and Wilfrid Hodges (2007). *Mathematical Logic*. Oxford University Press. ISBN 0199215626. 34
- Davis, Martin (1995). Influences of mathematical logic on computer science. In *The Universal Turing Machine A Half-Century Survey*, pages 289–299. Springer. 10
- Devlin, Keith (1992). *Sets, Functions and Logic*. Chapman & Hall, second edition. ISBN 0-412-45980-9. 48
- Feynman, Richard P; Robert B Leighton; and Matthew Sands (2011). *Feynman Lectures on Physics*. Basic Books, the new millenium edition edition. ISBN 0465023827. URL <http://www.feynmanlectures.caltech.edu/>. 10
- Halpern, Joseph Y; Robert Harper; Neil Immerman; Phokion G Kolaitis; Moshe Y Vardi; and Victor Vianu (2001). On the unusual effectiveness of logic in computer science. *Bulletin of Symbolic Logic*, pages 213–236. 10
- Hodges, Wilfred (1977). *Logic*. Penguin. ISBN 0140219854.
- Hodges, Wilfred (2001). *Logic*. Penguin, second edition. ISBN 0141003146.
- Huth, Michael and Mark Ryan (2004). *Logic in Computer Science: Modelling and Reasoning about Systems*. Cambridge University Press. ISBN 052154310X. 12
- Manna, Zohar (1974). *Mathematical Theory of Computation*. McGraw-Hill. ISBN 0-07-039910-7. 45, 48, 51, 52
- Mendelson, Elliott (1987). *Introduction to Mathematical Logic*. Wadsworth & Brooks/Cole, third edition. ISBN 0-534-06624-0. 45, 48
- Mendelson, Elliott (1997). *Introduction to Mathematical Logic*. Chapman & Hall, fifth edition. ISBN 1584888768. 51
- O'Donnell, John; Cordelia Hall; and Rex Page (2006). *Discrete Mathematics Using a Computer*. Springer, second edition. ISBN 1846282411. URL <http://www.dcs.gla.ac.uk/~jtod/discrete-mathematics/>. 39
- Polya, G. (1990). *How to Solve It: A New Aspect of Mathematical Method*. Penguin Books, second edition. Original first edition 1945 original second edition 1957.
- Smullyan, Raymond (1981). *What is the Name of This Book ?* Penguin. ISBN 0140223398. Original USA edition by Prentice-Hall 1978. 45, 46, 47, 48, 49, 50, 51
- Smullyan, Raymond (1984). *Alice in Puzzle-Land*. Penguin. Original edition by William Morrow 1982.
- Smullyan, Raymond (2000). *To Mock a Mockingbird: And Other Logic Puzzles*. Oxford Paperbacks. ISBN 0192801422. Original edition Knopf 1985, 1982.

- Smullyan, Raymond M (1983). *The Lady or the Tiger?: And Other Logic Puzzles*. Penguin. ISBN 0140224785. Original Knopf edition 1982.
- Smullyan, Raymond M. (1988). *Forever Undecided: A Puzzle Guide to Gödel*. OUP. ISBN 0192821962. Original Knopf edition 1987.
- Smullyan, Raymond M. (1990). *To Mock a Mockingbird: And Other Logic Puzzles*. Oxford Paperbacks. ISBN 0192801422. Original Knopf edition 1985; this is second Oxford edition (but no change), URL <http://revue.sesamath.net/IMG/pdf/mockingbird3.pdf>.
- Smullyan, Raymond M (1993). *Satan, Cantor and Infinity and Other Mind-boggling Puzzles*. Oxford Paperbacks. ISBN 0192861611. Original Knopf edition 1982.
- Smullyan, Raymond M. (1995). *First-Order Logic*. Dover Publications Inc. ISBN 0486683702.
- Smullyan, Raymond M and Melvin Fitting (2010). *Set Theory and the Continuum Problem*. Dover Publications. ISBN 0486474844. Original edition OUP 1996.
- Thompson, Simon (1991). *Type Theory and Functional Programming*. Addison Wesley. ISBN 0201416670. URL <http://www.cs.kent.ac.uk/people/staff/sjt/TTFP/>. 35
- Tunncliffe, Bob (1991). *Mathematics in Programming: An Introduction*. Prentice-Hall. ISBN 0135634040. 13
- Whitehead, Alfred North (1911). *An Introduction to Mathematics*. Williams and Norgate. 10

[Go to Table of Contents](#)