CSIT113
Problem Solving

Week 4

Some introductory logic

- Logic dates back to ancient Greece.
- There are several different ways of looking at logic but all have a common set of concepts.
  - Propositions:
    - Statements which are either True or False
  - Axioms:
    - Propositions which are True by definition
  - Theorems:
    - Propositions which can be proved to be True

Truth tables

- A useful tool in understanding logic is the truth table.
- This sets out all possible results in tabular form
- Logical operators (~, Λ, V, $\propto$, ≡) are often presented in truth tables.

Not (~ or ¬)

- The Not operator affects a single proposition, P.

P	~P
T	F
F	T

- Not reverses the truth value of P

And (∧) : conjunction

- The And operator combines two propositions, P and Q.
- And is true only if both P and Q are true

P	Q	P ∧ Q
T	T	T
T	F	F
F	T	F
F	F	F

Logical And

- Commutative: $\mathbf{p} \land \mathbf{q} \equiv \mathbf{q} \land \mathbf{p}$
- Associative: $\mathbf{p} \land (\mathbf{q} \land \mathbf{r}) \equiv (\mathbf{p} \land \mathbf{q}) \land \mathbf{r}$
- Idempotent: $\mathbf{p} \land \mathbf{p} \equiv \mathbf{p}$
- Has neutral element **`true`**: $\mathbf{p} \land \mathbf{true} \equiv \mathbf{p}$
- Distributes over boolean equality:
  $\mathbf{p} \land (\mathbf{q} \equiv \mathbf{r}) \equiv (\mathbf{p} \land \mathbf{q}) \equiv (\mathbf{p} \land \mathbf{r})$

Or (V) : disjunction

- The Or operator combines two propositions, P and Q.
- Or is true as long as at least one of P or Q is true

P	Q	P V Q
T	T	T
T	F	T
F	T	T
F	F	F

Logical Or

- Commutative: $\mathbf{p} \lor \mathbf{q} \equiv \mathbf{q} \lor \mathbf{p}$
- Associative: $\mathbf{p} \lor (\mathbf{q} \lor \mathbf{r}) \equiv (\mathbf{p} \lor \mathbf{q}) \lor \mathbf{r}$
- Idempotent: $\mathbf{p} \lor \mathbf{p} \equiv \mathbf{p}$
- Has neutral element **`false`**: $\mathbf{p} \lor \texttt{false} \equiv \mathbf{p}$
- Distributes over boolean equality:
  $\mathbf{p} \lor (\mathbf{q} \equiv \mathbf{r}) \equiv (\mathbf{p} \lor \mathbf{q}) \equiv (\mathbf{p} \lor \mathbf{r})$

Implies ($\propto$)   If-then …

- The Implies operator combines two propositions, P and Q.

P	Q	P $\propto$ Q
T	T	T
T	F	F
F	T	T
F	F	T

- Implies is true unless P is true and Q is false.
- This depends on the idea that if the if part (P) is false, then the overall cannot be false, so P $\propto$ Q would be considered true by default since it
  - Implies is misleading.

Some useful formulæ

- Implies (if-then)

  $\mathbf{P} \propto \mathbf{Q} \equiv \sim\mathbf{P} \vee Q$

P	Q	P ∝ Q	~P	~P V Q
T	T	T	F	T
T	F	F	F	F
F	T	T	T	T
F	F	T	T	T

- De Morgan’s laws

  $\sim(\mathbf{p} \vee \mathbf{q}) \equiv \sim\mathbf{p} \wedge \sim\mathbf{q}$

  $\sim(\mathbf{p} \wedge \mathbf{q}) \equiv \sim\mathbf{p} \vee \sim\mathbf{q}$

P	Q	P V Q	~(P V Q)	~P	~Q	~PΛ~Q
T	T	T	F	F	F	F
T	F	T	F	F	T	F
F	T	T	F	T	F	F
F	F	F	T	T	T	T

Equivalence (≡)

- The Equivalence operator combines two propositions, P and Q.

P	Q	P ≡ Q
T	T	T
T	F	F
F	T	F
F	F	T

- Equivalence is true if P and Q have the same truth value

Testing propositions

- We can use truth tables to test propositions to determine whether they are theorems.
- E.g. $P \propto (Q \propto P)$

P	Q	$Q \propto P$	$P \propto (Q \propto P)$
T	T	T	T
T	F	T	T
F	T	F	T
F	F	T	T

- Because the last column is all true, $P \propto (Q \propto P)$ is a theorem

Simplifying Logic

- We can express all possible logical operators in terms of just two operators:
  - Not and And
  - Not and Or
- Consider $\sim(\sim P \land \sim Q)$

P	Q	~P	~Q	~P Λ ~Q	~(~P Λ ~Q)
T	T	F	F	F	T
T	F	F	T	F	T
F	T	T	F	F	T
F	F	T	T	T	F

- Which is $P \lor Q$

Even simpler

- The two operators Nor and Nand make life even easier…

P	Q	P nor Q
T	T	F
T	F	F
F	T	F
F	F	T

P	Q	P Nand Q
T	T	F
T	F	T
F	T	T
F	F	T

- Either of these can produce all operators on its own…

Nor

- P nor P is the same as ~P
- ~P nor ~Q is the same as P ∧ Q
    - (P nor P) nor (Q nor Q)
- ~(P nor Q) is the same as P ∨ Q
    - (P nor Q) nor (P nor Q)

Knights and Knaves

- Every inhabitant of a mythical island is either a knight or a knave.
- Knights always tell the truth.
- Knaves always lie.
- This forms the basis of several problems in logic.

Problem 1

- It is rumoured there is gold on the island.
- A native tells you “The statement ‘there is gold on the island’ and the statement ‘I am a knight’ are either both true or both false”.
    - Can you tell if the native is a knight?
    - Can you tell if there is gold on the island?

Problem 2

- You come across two natives.
- You ask each if the other is a knight.
    - Do you get the same answer from both of them?

Problem 3

- There are three natives, A, B and C.
- A says “B and C are of the same type”.
  - What can we conclude about the number of knights present?

Problem 4

- There are three natives, A, B and C.
- A says “B and C are of the same type”.
  - What question can we ask C to find out if A is telling the truth?

Problem 5

- There are two natives, A and B.
  - What question should you ask A to determine if B is a knight?

Problem 6

- There are two natives, A and B.
  - What question should you ask A to determine whether A and B are of the same type?

Problem 7

- You come to a fork in the road.
- There is a restaurant down one of the two branches.
- There is a native at the fork.
    - What question do you ask to find out if the restaurant lies down the left fork?

Brute Force

- It is tempting to try to solve these problems by looking at all possible cases.
- The problems here are:
  - the number of cases rapidly becomes too large
  - the answer is often still not clear
- Perhaps there is a better technique.
- One approach is *Calculational Logic*

Calculational Logic

- The basis of calculational logic is to calculate with Boolean expressions.
- These expressions, called propositions, are either true or false.

Boolean Equality

- The Boolean equality relation satisfies a number of properties:
  - Reflexive: $\mathbf{p} \equiv \mathbf{p}$
  - Symmetric: $(\mathbf{p} \equiv \mathbf{q}) \equiv (\mathbf{q} \equiv \mathbf{p})$.
  - Transitive: if $\mathbf{p} \equiv \mathbf{q}$ and $\mathbf{q} \equiv \mathbf{r}$ then $\mathbf{p} \equiv \mathbf{r}$.
  - Associative: $\mathbf{p} \equiv (\mathbf{q} \equiv \mathbf{r}) \equiv (\mathbf{p} \equiv \mathbf{q}) \equiv \mathbf{r}$
  - Substitution of equals for equals:
    if $\mathbf{p} \equiv \mathbf{q}$ and f is a Boolean function then $f(\mathbf{p}) \equiv f(\mathbf{q})$.

Knights and Knaves

- If A is a native of the island the statement “A is a knight” is either true or false.
- So, the statement is a proposition.
- Let **A** represent the proposition “A is a knight”.
- Suppose A makes some statement **S**.
- The truth or falsity of this statement is the same as the truth or falsity of **A**.

  $\mathbf{A} \equiv \mathbf{S}$

Knights and Knaves

- So if A says “the restaurant is to the left” then **A ≡ L**.
- In other words either A is a knight and the restaurant is to the left or A is not a knight and the restaurant is not to the left.
- If A says “I am a knight” we conclude that **A ≡ A** which tells us nothing!
    - Everyone claims to be a knight.

Knights and Knaves

- If we ask A a Yes/No question, Q, the response will be the truth value of $\mathbf{A} \equiv \mathbf{Q}$.
- That is, if the response is “yes”, either A is a knight and the answer to Q really is yes or A is a knave and the answer is really no.
- Otherwise the response will be “no”.

Knights and Knaves

- Let's say we have two natives, A and B.
- A says “B is a knight”
  - What can we deduce?
- If **A** represents the proposition A is a knight and **B** represents the proposition B is a knight:

  $\mathbf{A} \equiv \mathbf{B}$.
- That is, A and B are of the same type.
- Note that we don’t know *which* type.

Problem 1

- It is rumoured there is gold on the island.
- A native tells you “The statement ‘there is gold on the island’ and the statement ‘I am a knight’ are either both true or both false”.
  - Can you tell if the native is a knight?
  - Can you tell if there is gold on the island?

Problem 1

- If A says "The statement 'there is gold on the island' and the statement 'I am a knight' are either both true or both false" he is asserting $\mathbf{A} \equiv \mathbf{G}$ where $\mathbf{A}$ is the assertion A is a knight and $\mathbf{G}$ the assertion there is gold on the Island.
- Any assertion by a native has the same truth value as $\mathbf{A}$ so:

$$\mathbf{A} \equiv (\mathbf{A} \equiv \mathbf{G})$$

$$(\mathbf{A} \equiv \mathbf{A}) \equiv \mathbf{G}$$

$$\mathtt{true} \equiv \mathbf{G}$$

Problem 1

- From this we can conclude that there is gold on the island, even though we have no idea if the native is a knight or a knave.

Problem 2

- You come across two natives.
- You ask each if the other is a knight.
  - Do you get the same answer from both of them?

Problem 2

- A will answer “yes” if he is a knight and so is B or if he is a knave and so is B.
- In other words:
  - $Q \equiv (A \equiv B)$
- Using the symmetry property:
  - $(A \equiv B) \equiv (B \equiv A)$
  - $Q \equiv (B \equiv A)$
- So B’s answer will be the same as A’s.

Problem 3

- There are three natives, A, B and C.
- A says “B and C are of the same type”.
  - What can we conclude about the number of knights present?

Problem 3

- A says **B** ≡ **C** so:
  - **A** ≡ (**B** ≡ **C**)
- So
  - A is a knight and so are B and C

  or
  - A is a knight and B and C are knaves

  or
  - A is a knave and one of B and C is a knight
- There is an odd number of knights.

Problem 4

- There are three natives, A, B and C.
- A says “B and C are of the same type”.
    - What question can we ask C to find out if A is telling the truth?

Problem 4

- Let Q be the unknown question we must ask, with truth value **Q**.
- Let **A**, **B** and **C** denote the propositions A, B, C is a knight.
- The response we want is **A** so:
  - $(Q \equiv C) \equiv A$
- Which we regroup to give:
  - $Q \equiv (C \equiv A)$
- But $A \equiv (B \equiv C)$ so substituting for A we get:
  - $Q \equiv (C \equiv (B \equiv C))$
- Which simplifies (after rearrangement) to:
  - $Q \equiv (C \equiv (B \equiv C))$
  - $Q \equiv (C \equiv (C \equiv B))$
  - $Q \equiv ((C \equiv C) \equiv B)$
  - $Q \equiv (\text{true} \equiv B)$
- So you ask "Is B a knight?"

Problem 5

- There are two natives, A and B.
  - What question should you ask A to determine if B is a knight?

Problem 5

- We want a question, Q, whose answer, when asked of A, is the type of B.
  - (**Q** ≡ **A**) ≡ **B**
- Reorganising:
  - **Q** ≡ (**A** ≡ **B**)
- In other words “Is B of the same type as you?”.

Problem 6

- There are two natives, A and B.
  - What question should you ask A to determine whether A and B are of the same type?

Problem 6

- We want a question, Q, which, when asked of A, determines if A and B are of the same type:
  - $(\mathbf{Q} \equiv \mathbf{A}) \equiv (\mathbf{A} \equiv \mathbf{B})$
- Regrouping and simplifying:
  - $\mathbf{Q} \equiv (\mathbf{A} \equiv (\mathbf{A} \equiv \mathbf{B}))$
  - $\mathbf{Q} \equiv ((\mathbf{A} \equiv \mathbf{A}) \equiv \mathbf{B})$
  - $\mathbf{Q} \equiv (\mathtt{true} \equiv \mathbf{B})$
- In other words “Is B a knight?”.

Problem 7

- You come to a fork in the road.
- There is a restaurant down one of the two branches.
- There is a native at the fork.
  - What question do you ask to find out if the restaurant lies down the left fork?

Problem 7

- Following the same rules as before:
  - $(Q \equiv A) \equiv L$
- Which we can rearrange as:
  - $Q \equiv (A \equiv L)$
- So our question is “Is the truth value of the statement ‘you are a knight’ the same as the truth value of ‘the restaurant lies down the left fork’?”
- Or in simple English “Would you say the restaurant is down the left fork?”

“Would you say the restaurant is down the left fork?”

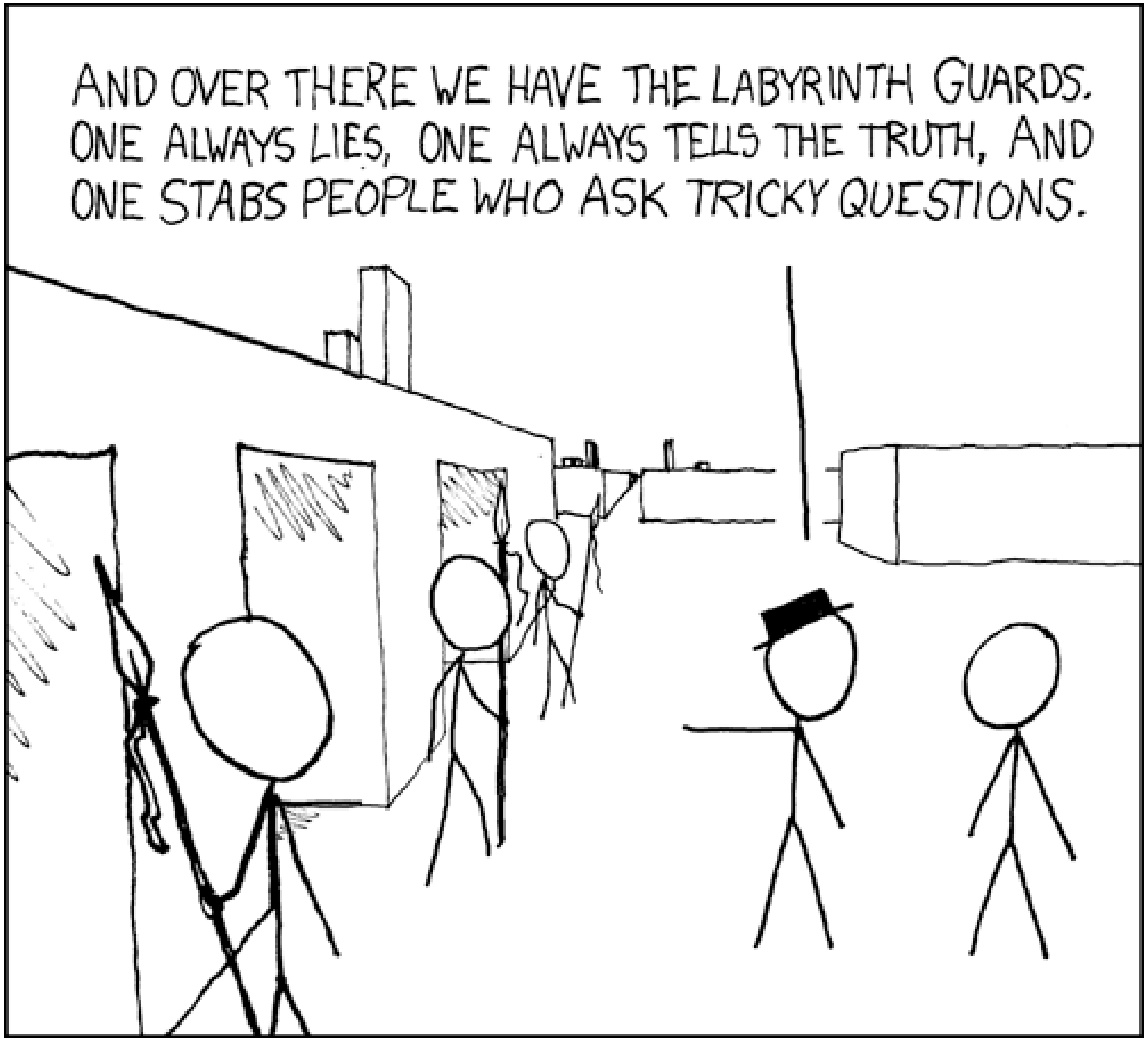
- We can check this is correct using brute force ☺
  - A’s answer to Q should be the same as L

A	L	L*	Q
T	T	T	T
T	F	F	F
F	T	F	T
F	F	T	F

  - Note: L* is the answer A would give to the question “is the restaurant down the left fork”

The Liar's Paradox

- "*This statement is false.*"
- Actually this version can be resolved by reinterpreting what true and false mean, and saying something can be neither true nor false.
  - Three valued logics
- This leads to "*This statement is not true.*"
  - The neither true nor false escape clause doesn't work now!
  - However, what if it is true and false?
    - This is allowed in some *paraconsistent* logics.
- *This statement is only false.*
  - Now we are in real trouble!



http://xkcd.com/246/