

## Introduction

- "Weak" (search-based) problem-solving does not scale to real problems.
- To succeed, problem solving needs *domain specific knowledge*.
- In search, knowledge = heuristic.
- We need to be able to represent knowledge efficiently.
- One way to do this is to use logic.

```
cis32-fall2005-parsons-lect13
```

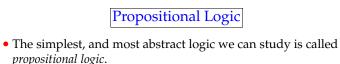
## What is a Logic?

- When most people say 'logic', they mean either *propositional logic* or *first-order predicate logic*.
- However, the precise definition is quite broad, and literally hundreds of logics have been studied by philosophers, computer scientists and mathematicians.
- Any 'formal system' can be considered a logic if it has:
  - a well-defined *syntax*;
  - a well-defined *semantics*; and
  - a well-defined *proof-theory*.

- The *syntax* of a logic defines the syntactically acceptable objects of the language, which are properly called *well-formed formulae* (wff). (We shall just call them formulae.)
- The *semantics* of a logic associate each formula with a *meaning*.
- The *proof theory* is concerned with manipulating formulae according to certain rules.

cis32-fall2005-parsons-lect13

3



- **Definition:** A *proposition* is a statement that can be either *true* or *false;* it must be one or the other, and it cannot be both.
- EXAMPLES. The following are propositions:
  - the reactor is on;
  - the wing-flaps are up;
  - Marvin K Mooney is president.
  - whereas the following are not:
  - are you going out somewhere?

-2+3

#### cis32-fall2005-parsons-lect13

- Now, rather than write out propositions in full, we will abbreviate them by using *propositional variables*.
- It is standard practice to use the lower-case roman letters

 $p, q, r, \ldots$ 

to stand for propositions.

• If we do this, we must define what we mean by writing something like:

Let *p* be Marvin K Mooney is president.

• Another alternative is to write something like *reactor\_is\_on*, so that the interpretation of the propositional variable becomes obvious.

• It is possible to determine whether any given statement is a proposition by prefixing it with:

*It is true that ...* 

and seeing whether the result makes grammatical sense.

- We now define *atomic* propositions. Intuitively, these are the set of smallest propositions.
- **Definition:** An *atomic proposition* is one whose truth or falsity does not depend on the truth or falsity of any other proposition.
- So all the above propositions are atomic.

cis32-fall2005-parsons-lect13

## The Connectives

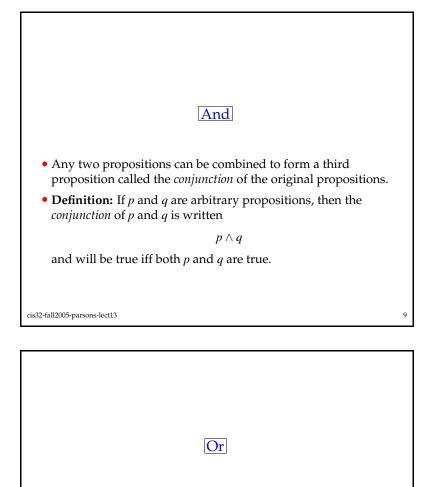
- Now, the study of atomic propositions is pretty boring. We therefore now introduce a number of *connectives* which will allow us to build up *complex propositions*.
- The connectives we introduce are:

 $\land \text{ and (\& or .)}$  $\lor \text{ or (| or +)}$  $\neg \text{ not (~)}$  $\Rightarrow \text{ implies ($\supset$ or $\rightarrow$)}$  $\Leftrightarrow \text{ iff ($\leftrightarrow$)}$ 

• (Some books use other notations; these are given in parentheses.)

cis32-fall2005-parsons-lect13

7



- Any two propositions can be combined by the word 'or' to form a third proposition called the *disjunction* of the originals.
- **Definition:** If *p* and *q* are arbitrary propositions, then the *disjunction* of *p* and *q* is written

cis32-fall2005-parsons-lect13

 $p \vee q$ 

and will be true iff either p is true, or q is true, or both p and q are true.

- We can summarise the operation of ∧ in a *truth table*. The idea of a truth table for some formula is that it describes the behaviour of a formula under all possible interpretations of the primitive propositions the are included in the formula.
- If there are *n* different atomic propositions in some formula, then there are 2<sup>*n*</sup> different lines in the truth table for that formula. (This is because each proposition can take one 1 of 2 values *true* or *false*.)
- Let us write *T* for truth, and *F* for falsity. Then the truth table for  $p \land q$  is:

| p | q | $p \wedge q$ |
|---|---|--------------|
| F | F | F            |
| F | Т | F            |
| Т | F | F            |
| Т | Т | Т            |
|   |   |              |
|   |   |              |

10

12

• The operation of  $\lor$  is summarised in the following truth table:

| р | q | $p \lor q$ |
|---|---|------------|
| F | F | F          |
| F | Т | Т          |
| Т | F | Т          |
| Т | Т | Т          |

• Note that this 'or' is a little different from the usual meaning we give to 'or' in everyday language.

cis32-fall2005-parsons-lect13

11

| If Then         • Many statements, particularly in mathematics, are of the form: <i>if</i> p <i>is true then</i> q <i>is true.</i> Another way of saying the same thing is to write:         p <i>implies</i> q.         • In propositional logic, we have a connective that combines two propositions into a new proposition called the <i>conditional</i> , or <i>implication</i> of the originals, that attempts to capture the sense of such a statement. |   |
|---|---|
| <ul> <li><i>if</i> p <i>is true then</i> q <i>is true.</i></li> <li>Another way of saying the same thing is to write:<br/>p <i>implies</i> q.</li> <li>In propositional logic, we have a connective that combines two propositions into a new proposition called the <i>conditional</i>, or <i>implication</i> of the originals, that attempts to capture the sense of such a statement.</li> </ul>   | If Then   |
| <ul> <li>Another way of saying the same thing is to write:<br/>p <i>implies</i> q.</li> <li>In propositional logic, we have a connective that combines two propositions into a new proposition called the <i>conditional</i>, or <i>implication</i> of the originals, that attempts to capture the sense of such a statement.</li> </ul>  | • Many statements, particularly in mathematics, are of the form:  |
| <ul> <li>p <i>implies</i> q.</li> <li>In propositional logic, we have a connective that combines two propositions into a new proposition called the <i>conditional</i>, or <i>implication</i> of the originals, that attempts to capture the sense of such a statement.</li> </ul>  | <i>if</i> p <i>is true then</i> q <i>is true.</i>   |
| • In propositional logic, we have a connective that combines two propositions into a new proposition called the <i>conditional</i> , or <i>implication</i> of the originals, that attempts to capture the sense of such a statement.  | Another way of saying the same thing is to write:   |
| propositions into a new proposition called the <i>conditional</i> , or <i>implication</i> of the originals, that attempts to capture the sense of such a statement.   | p <i>implies</i> q.   |
| 32-fall2005-parsons-lect13 13   | propositions into a new proposition called the <i>conditional</i> , or <i>implication</i> of the originals, that attempts to capture the sense of |
|   | 2-fall2005-parsons-lect13 1   |
|   |   |
|   |   |
|   |   |
|   |   |

• The  $\Rightarrow$  operator is the hardest to understand of the operators we

have considered so far, and yet it is extremely important.

• If you find it difficult to understand, just remember that the

If *p* is false, then we don't care about *q*, and by default, make

• Terminology: if  $\phi$  is the formula  $p \Rightarrow q$ , then *p* is the *antecedent* of

 $p \Rightarrow q$  means 'if p is true, then q is true'.

 $p \Rightarrow q$  evaluate to *T* in this case.

 $\phi$  and q is the consequent.

• **Definition:** If *p* and *q* are arbitrary propositions, then the *conditional* of *p* and *q* is written

#### $p \Rightarrow q$

and will be true iff either p is false or q is true.

• The truth table for  $\Rightarrow$  is:

| р | q | $p \Rightarrow q$ |
|---|---|-------------------|
| F | F | Т                 |
| F | Т | Т                 |
| Т | F | F                 |
| Т | Т | Т                 |

14

16

cis32-fall2005-parsons-lect13

Iff

• Another common form of statement in maths is:

p is true if, and only if, q is true.

- The sense of such statements is captured using the *biconditional* operator.
- **Definition:** If *p* and *q* are arbitrary propositions, then the *biconditional* of *p* and *q* is written:

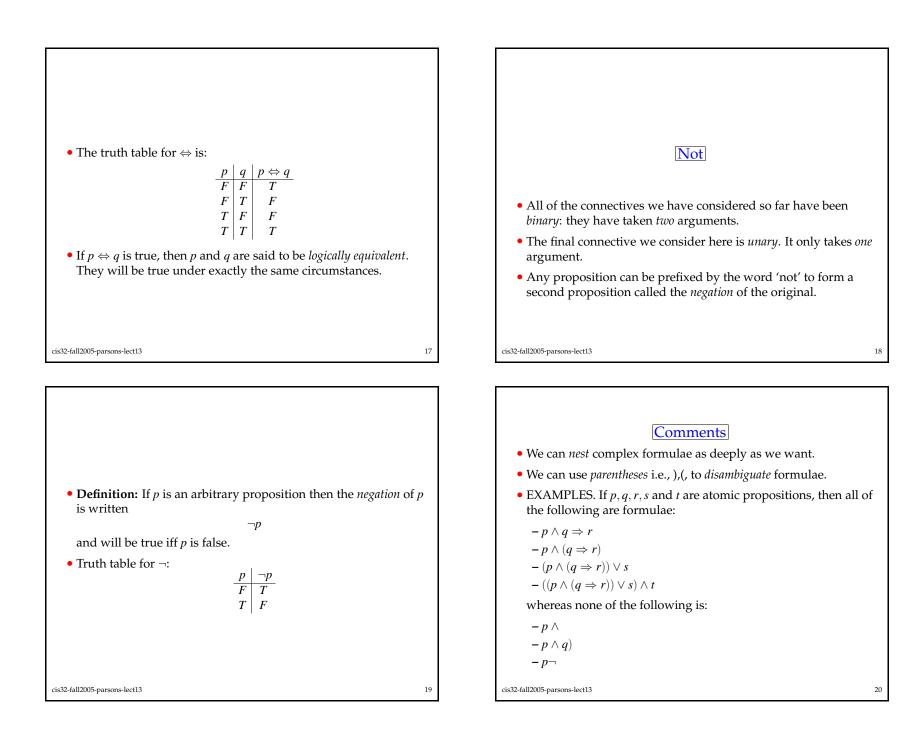
 $p \Leftrightarrow q$ 

and will be true iff either:

- 1. *p* and *q* are both true; or
- 2. *p* and *q* are both false.

cis32-fall2005-parsons-lect13

15



## Tautologies & Consistency

- Given a particular formula, can you tell if it is true or not?
- No you usually need to know the truth values of the component atomic propositions in order to be able to tell whether a formula is true.
- **Definition:** A *valuation* is a function which assigns a truth value to each primitive proposition.
- In Modula-2, we might write:

```
PROCEDURE Val(p : AtomicProp):
BOOLEAN;
```

- Given a valuation, we can say for any formula whether it is true or false.
- A valuation is also known as an *interpretation*

cis32-fall2005-parsons-lect13

- When we consider formulae in terms of interpretations, it turns out that some have interesting properties.
- Definition:
  - 1. A formula is a *tautology* iff it is true under *every* valuation;
  - 2. A formula is *consistent* iff it is true under *at least one* valuation;
  - 3. A formula is *inconsistent* iff it is not made true under *any* valuation.
- A tautology is said to be *valid*.
- A consistent formula is said to be *satisfiable*.
- An inconsistent formula is said to be *unsatisfiable*.

• EXAMPLE. Suppose we have a valuation *v*, such that:

v(p) = F v(q) = Tv(r) = F

Then we truth value of  $(p \lor q) \Rightarrow r$  is evaluated by:

$$(v(p) \lor v(q)) \Rightarrow v(r) \tag{1}$$

$$= (F \lor T) \Rightarrow F \tag{2}$$

$$=T\Rightarrow F$$
 (3)

$$=F$$
 (4)

22

24

Line (3) is justified since we know that  $F \lor T = T$ . Line (4) is justified since  $T \Rightarrow F = F$ . If you can't see this, look at the truth tables for  $\lor$  and  $\Rightarrow$ .

cis32-fall2005-parsons-lect13

21

23

- **Theorem:**  $\phi$  is a tautology iff  $\neg \phi$  is unsatisfiable.
- Now, each line in the truth table of a formula corresponds to a valuation.
- So, we can use truth tables to determine whether or not formulae are tautologies.
- If every line in the truth tabel has value *T*, the the formula is a tautology.
- Also use truth-tables to determine whether or not formulae are *consistent*.

- To check for consistency, we just need to find *one* valuation that satisfies the formula.
- If this turns out to be the first line in the truth-table, we can stop looking immediately: we have a *certificate* of satisfiability.
- To check for validity, we need to examine *every* line of the truth-table.

No short cuts.

• The lesson? Checking satisfiability is easier than checking validity.

cis32-fall2005-parsons-lect13

• The primitive propositions will be used to represent statements such as:

I am in Brooklyn It is raining It is Friday 11th November 2005.

These are primitive in the sense that they are *indivisible*; we cannot break them into smaller propositions.

• The remaining logical connectives (∧, ⇒, ⇔) will be introduced as abbreviations.

## Syntax

- We have already informally introduced propositional logic; we now define it formally.
- To define the syntax, we must consider what symbols can appear in formulae, and the rules governing how these symbols may be put together to make acceptable formulae.
- **Definition:** Propositional logic contains the following symbols:
  - 1. A set of *primitive propositions*,  $\Phi = \{p, q, r \dots \}$ .
  - The unary logical connective '¬' (not), and binary logical connective '∨' (or).

26

28

3. The punctuation symbols ')' and '('.

cis32-fall2005-parsons-lect13

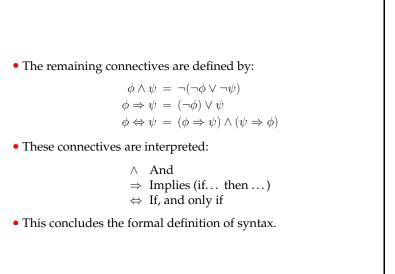
25

27

- We now look at the rules for putting formulae together.
- **Definition:** The set *W*, of (well formed) formulae of propositional logic, is defined by the following rules:
  - 1. If  $p \in \Phi$ , then  $p \in W$ . 2. If  $\phi \in W$ , then:

 $\neg \phi \in \mathcal{W}$  $(\phi) \in \mathcal{W}$ 

3. If  $\phi \in W$  and  $\psi \in W$ , then  $\phi \lor \psi \in W$ .



cis32-fall2005-parsons-lect13

- QUESTION: How can we tell whether a formula is  $\top$  or  $\perp$ ?
- For example, consider the formula

 $(p \land q) \Rightarrow r$ 

Is this  $\top$ ?

- The answer must be: *possibly*. It depends on your *interpretation* of the primitive propositions *p*, *q* and *r*.
- The notion of an interpretation is easily formalised.
- **Definition:** An *interpretation* for propositional logic is a function

 $\pi: \Phi \mapsto \{T, F\}$ 

which assigns T (true) or F (false) to every primitive proposition.

# Semantics • We now look at the more difficult issue of *semantics*, or *meaning*. • What does a proposition *mean*? • That is, when we write It is raining. what does it mean?

From the point of view of logic, this statement is a *proposition*: something that is either  $\top$  or  $\bot$ .

- *The meaning of a primitive proposition is thus either*  $\top$  *or*  $\perp$ *.*
- In the same way, the meaning of a formula of propositional logic is either  $\top$  or  $\perp$ .

30

32

cis32-fall2005-parsons-lect13

29

31

- But an interpretation only gives us the meaning of primitive propositions; what about complex propositions arbitrary formulae?
- We use some *rules* which tell us how to obtain the meaning of an arbitrary formulae, given some interpretation.
- Before presenting these rules, we introduce a symbol:  $\models$ . If  $\pi$  is an interpretation, and  $\phi$  is a formula, then the expression

```
\pi \models \phi
```

will be used to represent the fact that  $\phi$  is  $\top$  under the interpretation  $\pi$ .

Alternatively, if  $\pi \models \phi$ , then we say that:

–  $\pi$  satisfies  $\phi$ ; or

```
– \pi models \phi.
```

• The symbol  $\models$  is called the *semantic turnstile*.

cis32-fall2005-parsons-lect13

• The rule for primitive propositions is quite simple. If  $p \in \Phi$  then  $\pi \models p$  iff  $\pi(p) = T$ 

$$\pi \models p \inf \pi(p) = I$$

- The remaining rules are defined *recursively*.
- The rule for  $\neg$ :

$$\pi \models \neg \phi \text{ iff } \pi \not\models \phi$$

(where  $\not\models$  means 'does not satisfy'.)

• The rule for  $\lor$ :

$$\pi \models \phi \lor \psi$$
 iff  $\pi \models \phi$  or  $\pi \models \psi$ 

cis32-fall2005-parsons-lect13

- If we are given an interpretation  $\pi$  and a formula  $\phi$ , it is a simple (if tedious) matter to determine whether  $\pi \models \phi$ .
- We just apply the rules above, which eventually bottom out of the recursion into establishing if some proposition is true or not.

• So for:

 $(p \lor q) \land (q \lor r)$ 

we first establish if  $p \lor q$  or  $q \lor r$  are true and then work up to the compound proposition.

• Since these are the only connectives of the language, these are the only semantic rules we need.

Since:

 $\phi \Rightarrow \psi$ 

 $(\neg \phi) \lor \psi$ 

it follows that:

is defined as:

$$\pi \models \phi \Rightarrow \psi \text{ iff } \pi \not\models \phi \text{ or } \pi \models \psi$$

34

36

• And similarly for the other connectives we defined.

cis32-fall2005-parsons-lect13

33

35

## Summary

- This lecture started to look at logic from the standpoint of artificial intelligence.
- The main use of logic from this perspective is as a means of knowledge representation.
- We introduced the basics of propositional logic, and talked about some of the properties of sentences in logic.
- We also looked at a formal definition of syntax and semantics, and the semantic approach to inference.
- The next lecture will look at the syntactic approach—proof theory.