CS1112
The Structure of Logical Statements

Lecturer:

Professor Barry O'Sullivan

Office: 2-65, Western Gateway Building

email: *b.osullivan@cs.ucc.ie*

http://osullivan.ucc.ie/teaching/cs1112/

Understanding the structure of logical statements

The language of logic

The structure of complex statements

Building complex truth tables

Understanding Specifications

You software must ensure both of the following conditions:
(i)whenever the user is logged out, their balance is hidden
(ii)the balance cannot be visible at the same time as a transaction is being processed

1. How do we work out whether or not some given software satisfies the specification?
2. Can we work out the truth value for all possible situations?
3. If we write it as $p \rightarrow q \wedge \neg\neg q \wedge s$ does it make a difference?
4. What about $(p \vee s) \rightarrow q$?

p: the user is logged out
q: the user's balance is hidden
s: a transaction is being processed

$$((p \rightarrow q) \wedge (\neg(\neg q \wedge s)))$$

Understanding complex logic statements

- In order to understand complex statements, we have to work out how they are structured

- To do that, we have to understand the formal language
- We will define a set of well-formed formulae (or wff) using:
  - propositional symbols: $\{p, q, r, \ldots, p', p'', \ldots\}$
  - logical connectives: $\{\neg, \wedge, \vee, \rightarrow, \leftrightarrow\}$
  - punctuation: { (,) }

1. All propositional symbols on their own are wffs
2. if p and q are wffs, then so are
   (p), $(\neg p)$, $(p \wedge q)$, $(p \vee q)$, $(p \rightarrow q)$ and $(p \leftrightarrow q)$
3. nothing else is a wff unless it can be formed by repeatedly applying rules 1 and 2 above.

Examples

Well-formed

p
$(p \wedge q)$
$(p \rightarrow r)$
$((q \vee r) \rightarrow w)$
$(((p \wedge q) \wedge r) \wedge w)$
$(r \leftrightarrow (\neg q))$

Not well-formed

pq
$p \wedge \vee r$
$\rightarrow r$
$(q \wedge p(\leftrightarrow r))$
$r \neg \leftrightarrow q$

$q \vee r \rightarrow w$
$r \leftrightarrow \neg q$

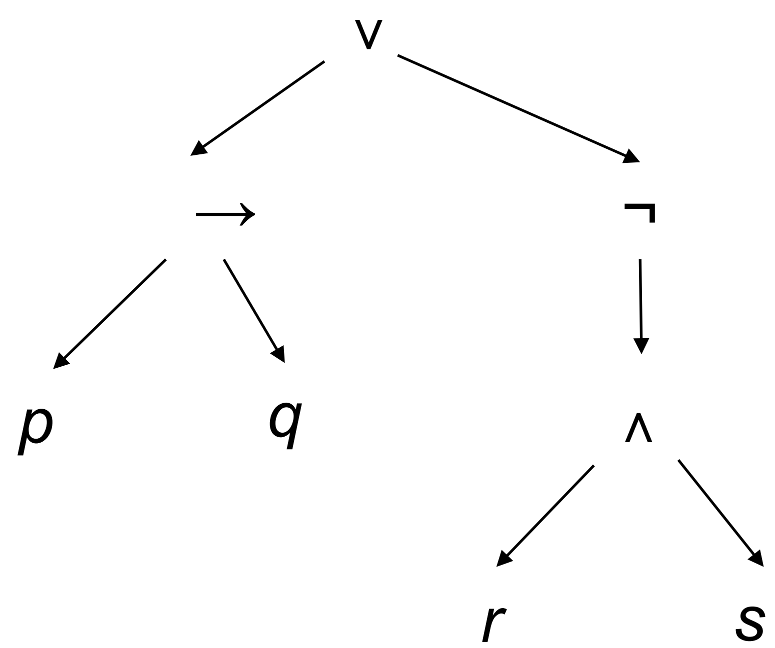
... but see later slides

Determining the structure of a wff

Each wff is either an atomic proposition, or formed by using a single connective, or is ¬ applied to a proposition, or is a connective linking two propositions.

We can use this to build a tree, in which the root is the main connective.

$((p \to q) \vee (\neg(r \wedge s)))$



Note on drawing trees

Each connective should have 1 or 2 branches coming out it

- 1 branch coming out of "¬"
- 2 branches coming out of "∧", "∨", "→" and "↔"

The branches must be directed down the page, and they point to the sub-wffs that the connective acts upon

A propositional symbol (e.g. *p*, *q*, etc.) never has a branch coming out of it.

Brackets (e.g. "(" or ")") never appear in the tree.

Do we need the brackets in the statement?

What would the structure tree be for the following formula?

$$q \vee r \wedge w$$

"You are allowed in this bar if: you are at least 18 years old or it is before 9pm and you are accompanied by a parent."

Omitting brackets

Often, we will drop some of the brackets for convenience, but we then need to say how to disambiguate the formula

We will apply the connectives in the following order:

$\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$
unless brackets tell us otherwise.

Note: this means apply the connectives to the closest matching propositions

This is important! Memorise this order, and use it.

In a sequence of $\wedge$, we will combine them on the left first. Similarly for $\vee$, $\rightarrow$ and $\leftrightarrow$

But in this module, we will try to use brackets in most cases, to avoid confusion.

Example

recreate wffs from the following

$\underline{p \vee q \wedge r \rightarrow \neg p}$

$p \vee q \wedge r \rightarrow (\neg p)$

$p \vee (q \wedge r) \rightarrow (\neg p)$

$(p \vee (q \wedge r)) \rightarrow (\neg p)$

$((p \vee (q \wedge r)) \rightarrow (\neg p))$

$\underline{p \vee q \vee r \vee s \wedge t \wedge w}$

$p \vee q \vee r \vee (s \wedge t) \wedge w$

$p \vee q \vee r \vee ((s \wedge t) \wedge w)$

$(p \vee q) \vee r \vee ((s \wedge t) \wedge w)$

$((p \vee q) \vee r) \vee ((s \wedge t) \wedge w)$

$(((p \vee q) \vee r) \vee ((s \wedge t) \wedge w))$

Exercise

$\neg, \wedge, \vee, \rightarrow, \leftrightarrow$

What are the structure trees for the following formulae?

1. $p \wedge q \rightarrow r$

2. $p \wedge \neg q \rightarrow r$

3. $p \rightarrow q \wedge r$

4. $p \rightarrow \neg q \wedge r$

5. $p \wedge (q \rightarrow r)$

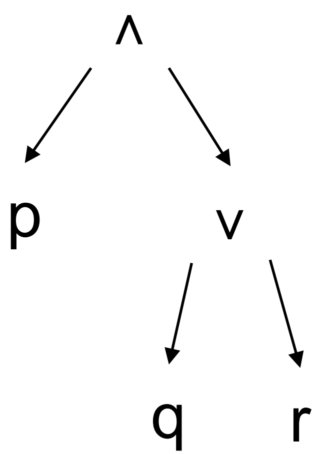
6. $(p \rightarrow q) \wedge r$

7. $\neg (p \rightarrow q) \wedge r$

Truth Functions for larger statements

We can apply the same tables to build truth functions for more complex statements

E.g. $p \wedge (q \vee r)$



all propositional symbols appearing in the statement

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

all possible combinations

X	Y	$X \vee Y$
T	T	T
T	F	T
F	T	T
F	F	F

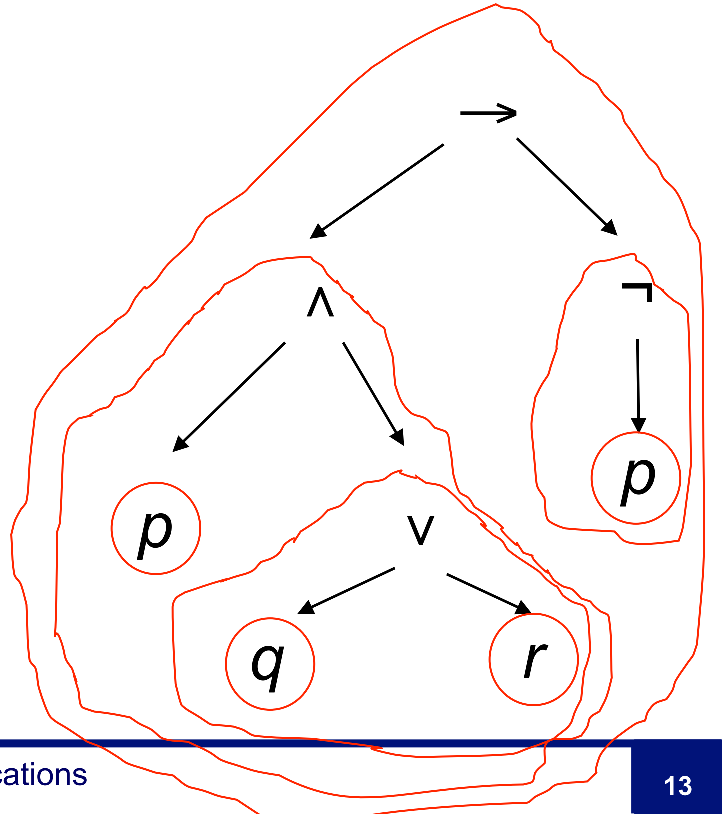
X	Y	$X \wedge Y$
T	T	T
T	F	F
F	T	F
F	F	F

Complex truth tables

$$(p \wedge (q \vee r)) \to \neg p$$

a	b	$a \to b$
T	T	T
T	F	F
F	T	T
F	F	T

p	q	r	$q \vee r$	$p \wedge (q \vee r)$	$\neg p$	$(p \wedge (q \vee r)) \to \neg p$
T	T	T	T	T	F	F
T	T	F	T	T	F	F
T	F	T	T	T	F	F
T	F	F	F	F	F	T
F	T	T	T	F	T	T
F	T	F	T	F	T	T
F	F	T	T	F	T	T
F	F	F	F	F	T	T



The truth of complex propositional statements

- we now have a method for deciding the truth of any complex propositional statement:
  1. work out how to read the expression
  2. construct the truth table
  3. determine the truth of the atomic propositions
  4. read the appropriate row in the table to get the truth value

- or
  1. work out how to read the expression
  2. determine the truth values of the atomic propositions
  3. construct the appropriate row of the truth table

Exercise

construct the truth table for $(q \rightarrow r) \vee p \wedge \neg r$

Next lecture ...

Logical equivalences
Satisfiability