

Lecture 9

Question Construct a wff $\text{XOR}(P, Q, R)$ which is T when exactly one of P, Q, R is T and F otherwise.

Attempt 1 What about $P \oplus (Q \oplus R)$?

We can check our answer using truth tables

Let $P = T, Q = T, R = T$.

Then $Q \oplus R = F \text{ so } P \oplus (Q \oplus R) = \underline{\underline{T}}$

Thus, this wff does not satisfy the conditions of the question.

Attempt 2 $(P \vee Q \vee R) \wedge \neg(P \wedge Q) \wedge \neg(P \wedge R)$
 $\wedge \neg(Q \wedge R)$

Check this wff by the truth table generator, and see that it works.

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

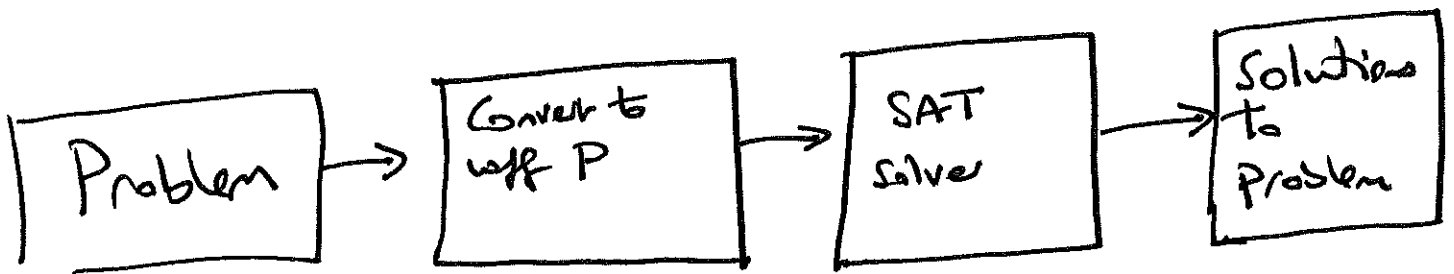
It is possible to generalize this result to obtain a wff that represents $\text{xor}(p_1, \dots, p_n)$ [which is T when exactly one of p_1, \dots, p_n is T and F on all other occasions].

Section 1.5: PL in action

We shall show (by means of an example) that PL can be used as a sort of a programming language. However, not in the way that say, Java, is a programming language. PL can be used to describe a problem/exactly ~~in a way that~~ by means of a well A in such a way that solutions to P correspond exactly to such assignments satisfying A.

More generally, we have to solve a
scheme for solving problems (which ones?)

using PL:

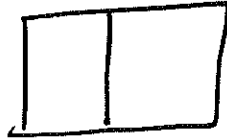


• P describes the problem — it doesn't solve it. (P does not represent an algorithm)

• Solutions to problem correspond to the truth assignments that satisfy P .

[Full-size Sudoku can be handled in this way (see exercise)].

Example We take a really simple
"Sudoku" puzzle



Just two cells. Each cell can contain exactly one of the digits 1 or 2.

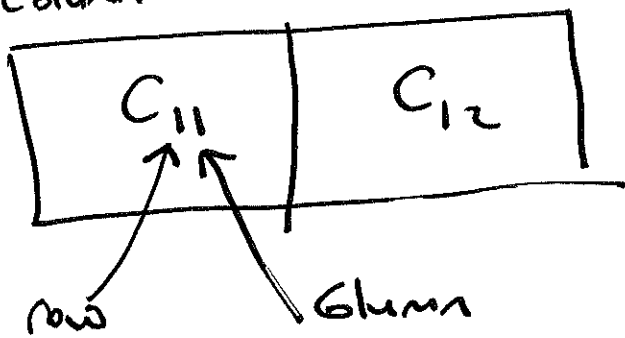
No number can be repeated in a row.

There are two obvious solutions to this puzzle:



I will now show how to represent the puzzle by means of a logic S , such that, where S is satisfiable corresponds exactly to the solutions to the puzzle. S describes the problem, it doesn't solve it. If describes constraints that constitute the problem. When S takes the value T corresponds exactly to a solution to the puzzle.

I first label the cells using ~~array~~ array
 (= matrix) notation, I am doing this
 because it will scale φ .



← row 1
 (in fact, the only
 row)

We shall need 4 atoms to describe
 what numbers can occur in which cells:

$P = 'C_{11} \text{ contains the number 1}'$

$Q = 'C_{11} \text{ contains the number 2}'$

$R = 'C_{12} \text{ contains the number 1}'$

$S = 'C_{12} \text{ contains the number 2}'$

Example The truth assignment

P	q	r	S
T	F	T	F

Corresponds to the following

1	1
---	---

The truth assignment

P	q	r	S
T	F	F	T

Corresponds to the following

1	2
---	---

We now need to describe the
constraints of ~~the~~ ^{this} Sudoku puzzle
 using only PL.

Constraints

$P \oplus Q$	This is T when C_{11} contains 1 or 2 but not both
$r \oplus S$	This is T when C_{12} contains 1 or 2 but not both
$P \oplus r$	This is T when C_{11} contains 1 or C_{12} contains 1 but not both
$Q \oplus S$	This is T when C_{11} contains 2 or C_{12} contains 2 but not both.

$$\text{Put } S = (P \oplus Q) \wedge (r \oplus S) \wedge (P \oplus r) \wedge (Q \oplus S).$$

The truth assignments to P, Q, r, S that make S
 T i.e. satisfy S correspond exactly to the
 solutions to the original Sudoku puzzle.

p	q	r	s	$(p \oplus q) \wedge (r \oplus s) \wedge (p \oplus r) \wedge (q \oplus s)$
T	T	T	T	F
T	T	T	F	F
T	T	F	T	F
T	T	F	F	F
T	F	T	T	F
T	F	T	F	F
T	F	F	T	T
(1)				
T	F	F	F	F
F	T	T	T	F
F	T	T	F	T
(2)				
F	T	F	T	F
F	T	F	F	F
F	F	T	T	F
F	F	T	F	F
F	F	F	T	F
F	F	F	F	F

Row (1) corresponds to $P=T, S=T, q=r=F$
 which is the solution

1	2
---	---

Row (2) corresponds to $P=F, q=T, r=T, s=F$
 which is the solution

2	1
---	---