A	B	A → B
T	T	T
T	F	F
F	T	T
F	F	T

when B is False A is negated so

¬A equiv to

A	A → ⊥
T	F
F	T

our version of (rewriting) ¬A

① $\neg A \Leftrightarrow A \to \perp$

COMP3620/COMP6320 Artificial Intelligence
Tutorial 3: Knowledge Representation and Reasoning

April 17–20, 2018

② $A \vee B$
$\Leftrightarrow \neg\neg A \vee B$
$\Leftrightarrow \neg A \to B$
$\Leftrightarrow (A \to \perp) \to B$

③ $A \wedge B$
$\Leftrightarrow \neg(\neg A \vee \neg B)$
$\Leftrightarrow \neg(A \to \neg B)$
$\Leftrightarrow \neg(A \to (B \to \perp))$
$\Leftrightarrow (A \to (B \to \perp)) \to \perp$

$A \to B$
$\Leftrightarrow \neg A \vee B$

Exercise 1

We know that disjunction ($\vee$), for instance, is definable in terms of conjunction ($\wedge$) and negation ($\neg$), since $A \vee B$ is equivalent to $\neg(\neg A \wedge \neg B)$.

Show that all of the usual connectives $\wedge$ ③, $\vee$ ② and $\neg$ ① are similarly definable in terms of implication ($\to$) and the false constant $\perp$.

Exercise 2

Consider the following set Γ of first order formulae:

$\Gamma = \{A, B, C\}$

$A \wedge B \wedge C$

$$\Gamma = \{ \forall x(\neg Q(x) \to P(x)), \quad \neg\exists y\, P(y), \quad Q(a) \to \exists x(R(x) \wedge \neg Q(x)) \}$$

$\Delta = ?$

$P \to Q$ is satisfiable b/c $\exists$ P = True, Q = True.

$P \wedge \neg P$ is not satisfiable b/c nothing will make it true.

* some assignment of truth values can make it true. $\Leftrightarrow$ satisfiable.

(a) Use normal-forming moves to transform Γ into a set Δ of first order clauses such that Δ is satisfiable if and only if Γ is satisfiable.

(b) Write out a resolution proof by which the empty clause is derived from Δ. For each resolution inference in the proof, make a note of any unifier that is involved.

(a). ① Prenex Normal Form
In each clause, the quantifiers should be at the front.
(the very first should be a quantifier)

$\neg(\exists y\, P(y)) \Leftrightarrow \forall y \neg P(y)$

$\boxed{\text{f.y.i. } \neg\forall x P(x) \Leftrightarrow \exists x \neg P(x)}$

$Q(a) \to \exists x(R(x) \wedge \neg Q(x))$

$\exists x[Q(a) \to R(x) \wedge \neg Q(x)]$

b/c there is no x on the LHS. so we can simply move it out safely.

② Remove $\exists$
$\exists x[Q(a) \to (R(x) \wedge \neg Q(x))]$
claim b is that x. b as a constant
$Q(a) \to (R(b) \wedge \neg Q(b))$

③ Remove $\forall$
just erase them b/c we know
letters like a, b, c ... constants
letters like x, y, z, ... variables
(implicit $\exists$ & $\forall$).
$\neg Q(x) \to P(x)$
$\neg P(y)$

④ negation directly to letters
$\neg(P \vee Q)$ ✗
$\neg P \vee Q$ ✓
also use $\wedge$ to connect all of them.

$A := \neg P(y)$ ✓

$B := \neg Q(x) \to P(x) \iff Q(x) \vee P(x)$ ✓

$C := Q(a) \to (R(b) \wedge \neg Q(b)) \iff \neg Q(a) \vee (R(b) \wedge \neg Q(b))$

$\iff (\neg Q(a) \vee R(b)) \wedge (\neg Q(a) \vee \neg Q(b))$ ✓

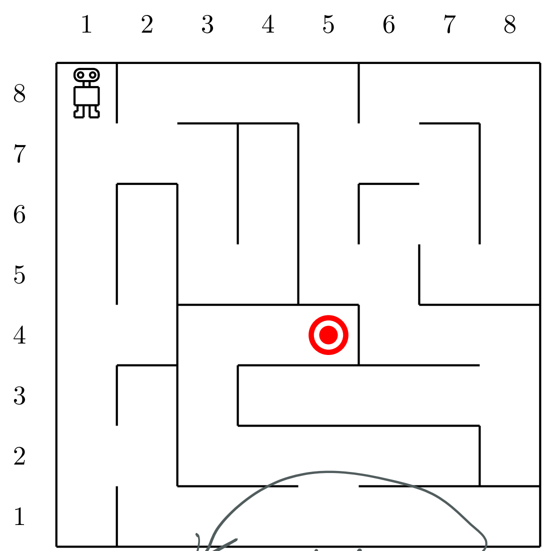
Exercise 3

$\Delta = A \wedge B \wedge C$ NORMALISED/CLAUSE FORM

The following is a simple gridworld problem in the form of a maze:

conjuncted form
$(\quad) \wedge (\quad) \cdots \wedge (\quad)$
↓
inside can have $\vee$ & $\neg$

AND OUTSIDE OR INSIDE



PATH: Series of steps
↓
True if step is valid
Last step ends at final square
first step starts at initial square

STATE
Robot's position
~~row, col → bool is the robot here?~~
row(state) = number
col(state) = number

ENVIRONMENT
List pairs of adjacent cells with a wall between them
bound the cell range

next(S)
next(next(S))

Say State S

next(S) 2 parts define it
① precondition ② result of it.

The robot in row 8, column 1 needs to find its way to the goal at row 4, column 5. The actions available to it in any given state are to move one square up, down, left or right—but it cannot exit the grid or walk through the walls.

Decide what is involved in describing this problem purely in terms of logic. The best approach is to think of some things you need to say, then see what vocabulary (predicates, function symbols, names) you need for that. The questions to be considered include:

- What objects are in the domain of discourse?
- What is invariant between states, and what vocabulary is required to describe it?
- What is required to define a specific state?
- How can we represent logically the relationship between an arbitrary state and its successor, and relate that to the possible actions?

Do not forget to include in the problem description the specification of the initial state and the goal.

For problems like this, it is convenient to use functions as well as relations. For example, the position of the robot in a state is a function of the state, not just a relation between the state and various places in the grid.

You can assume some basic properties of numbers, so you don't need to say specifically that 4 is 3+1, etc. You can also assume that the states are ordered, so for any state `s` (except maybe the last one) there is a next state `next(s)` for instance.

You do not have to write out every detail of the problem, especially as parts of it are a bit boring, but the closer you get to a full logical description the better. If you are happy with your encoding of this problem in terms of logic, and want to explore it further, just for fun, you may like to point your browser at
`https://l4f.cecs.anu.edu.au`
and in particular at
`https://l4f.cecs.anu.edu.au/puzzles/logician/traditional-maze`

RESULT
(After) next(s.right) $\to$ col(next(s,right)) = col(s)+1

PRECONDITIONS
$1 \le$ row(state) $\le$ row_max
$1 \le$ col(state) $\le$ col_max
next(s. right) $\to$ ~~(row(s), col(s)), (row(s), col(s)+1) ∉ walls~~
(Before) $\to \neg$ in_walls(row(s), col(s), (row(s), col(s)+1))

2.(b). unify x & y

by 2nd $\neg P(x)$, then for 1st $Q(x) \vee P(x)$

it must be $Q(x)$

1. $Q(x) \vee P(x)$
2. $\neg P(y)$
3. $Q(x)$ unify $(y \Leftarrow x)$
4. $\neg Q(a) \vee \neg Q(b)$ unify $(x \Leftarrow a)$
5. $\neg Q(b)$ (by 4)

6. $\perp$ unify $(x \Leftarrow b)$

$$(1 \wedge 2 \wedge 4) \wedge 3$$
$$\Downarrow$$
$$\perp \wedge 3$$
$$\Downarrow$$
$$\perp$$

empty clause