```latex
\documentclass[fullpage]{article}
\usepackage{graphicx}

% THIS FILE HAS TO BE RUN WITH PDFLATEX. IF YOU WANT TO RUN IT WITH LATEX AND YOU
HAVE THE FIGURE IN eps, YOU CAN
% UNCOMMENT THE LINES BELOW. HOWEVER, IT IS EASIER TO PRODUCE FIGURES IN PDF: YOU
CAN SEND ALMOST ANYTHING TO A THE "PDF PRINTER"
%\usepackage{epsfig}
%\usepackage{epstopdf}

\newcommand{\comlb}[1]{{\vspace{2mm}\noindent \bf REMARK (delete for
submission):}~ #1 \hfill {\bf
    END.}\\}

\title{\vspace*{-2cm}\bf Assignment 1 - COMP 3400}
\author{Khalil Van Alphen\\
Student Number: 100863992\\ Email: \  khalil.vanalphen@scs.carleton.ca}
\date{}
\begin{document}
\maketitle
\pagestyle{plain}
\thispagestyle{empty}

\noindent
1. Produce a knowledge base in PROPOSITIONAL LOGIC capturing
the following information as closely as possible as stated here: {\em If Tweety
is a bird and it is not an abnormal bird, then it flies. A bird is abnormal
exactly when it is an ostrich, a penguin, or not an abnormal wooden bird. A
wooden bird is abnormal exactly when operated under remote control. Tweety is a
bird, it is not an ostrich nor a penguin nor a remote controlled wooden bird.}
Prove using Prover9 that {\em Tweety flies}. Notice that the two forms of
abnormality mentioned here may be different, one is if birds, the other is for
wooden birds.\\

\noindent {\bf Solution:} \ We will use the propositional variables as listed
below:

\begin{itemize}
\item[$a$:]~ {\em Tweety is a bird}
\item[$b$:]~ {\em Tweety flies}
\item[$c$:]~ {\em A bird is abnormal}
\item[$d$:]~ {\em A bird is an ostrich}
\item[$e$:]~ {\em A bird is a penguin}
\item[$f$:]~ {\em A bird is an abnormal wooden bird}
\item[$g$:]~ {\em A bird is operated by remote control}
\item[$h$:]~ {\em A bird is wooden}
\end{itemize}
From this we are able to deduce a the following propositional formulas:
\begin{itemize}
\item[$\varphi_0$:]~ ${(a \wedge \neg c) \rightarrow b}$.

 Meaning: ``If Tweety is a bird and it is not an abnormal bird, then Tweety
flies."
\item[$\varphi_1$:]~ ${(d \lor e \lor \neg f) \rightarrow c}$
```

Meaning: ``If a bird is an ostrich, a penguin or not an abnormal wooden
bird, it is abnormal."
\item[φ_2:]~ ${(h \wedge g) \rightarrow f}$

 Meaning: ``If a bird is wooden and remote controlled, it is an abnormal wooden
bird."
\item[φ_3:]~ ${a}$

 Meaning: ``Tweety is a bird."
\item[φ_4:]~ ${\neg d}$

 Meaning: ``Tweety is not an ostrich."
\item[φ_5:]~ ${\neg e}$

 Meaning: ``Tweety is not a penguin."
\item[φ_6:]~ ${\neg h}$

 Meaning: ``Tweety is not wooden."
\item[φ_7:]~ ${\neg g}$

 Meaning: ``Tweety is not remote controlled."
\end{itemize}
We are attempting to prove that b is a consequence of $\varphi_0, \ldots,
\varphi_7$. Prover9 will accomplish this via transforming our formulas into
clauses and assumptions. Using prepositional logical equivalence, it will proceed
to recursively substitute terms with more atomic parts in order to get a singular
truth value. It will attempt to find a counterexample for the set of formula, and
will fail.

The main section of the attached input file is as follows:

{\small
\begin{verbatim}
    formulas(assumptions).

    (a & -c) -> b.
    (d | e | f) <-> c.
    (h & g) <-> f.
    a.
    -d.
    -e.
    -h.
    -g.

    end_of_list.

    formulas(goals).

    b.

    end_of_list.
 \end{verbatim}
}
 The main section of the attached output file is as follows:

```
  {\footnotesize \begin{verbatim}
    % Proof 1 at 0.03 (+ 0.00) seconds.
    % Length of proof is 15.
    % Level of proof is 4.
    % Maximum clause weight is 4.
    % Given clauses 0.

    1 a & -c -> b # label(non_clause).  [assumption].
    2 d | e | f <-> c # label(non_clause).  [assumption].
    3 h & g <-> f # label(non_clause).  [assumption].
    4 b # label(non_clause) # label(goal).  [goal].
    5 -a | c | b.  [clausify(1)].
    9 d | e | f | -c.  [clausify(2)].
    11 h | -f.  [clausify(3)].
    13 a.  [assumption].
    14 -d.  [assumption].
    15 -e.  [assumption].
    16 -h.  [assumption].
    18 -b.  [deny(4)].
    19 c.  [back_unit_del(5),unit_del(a,13),unit_del(c,18)].
    20 f.  [back_unit_del(9),unit_del(a,14),unit_del(b,15),unit_del(d,19)].
    21 $F.  [back_unit_del(11),unit_del(a,16),unit_del(b,20)].
\end{verbatim} }
```

As is clearly seen, the simple clausifications done in steps 5 through 11 make it
simple to substitute the values of the variables into the simpler formulas. By
taking the assumptions from 13 through 18 and the attempt to deny at 19, the
proof is only a matter of evaluating the clauses using logical operands.\\

\noindent
2. \ Prove with Prover9 the theorem about idempotency in Boolean algebras shown
in Chapter 2. You need to use predicate logic with ad hoc symbols for the
operations and special constants involved. Make sure Prover9 understands
equality. Instructions as for the previous problem.

More precisely, prove that {\em ``For all \ a:~~~ $a \sqcup a
= a$''} \ from the following axioms:

\begin{itemize}
\item [{BA1}] ~~Commutativity~~~~ For all~ a, b:
\begin{center}
$a \sqcup b = b \sqcup a$~~~~~~ $a \sqcap b = b \sqcap a$
\end{center}

\item [{BA2}] ~~Associativity:~~~ For all~ a, b, c:
\begin{center}
$(a \sqcup b) \sqcup c = a \sqcup (b \sqcup c)$\\ $(a
\sqcap b) \sqcap c = a \sqcap (b \sqcap c)$
\end{center}
\item [{BA3}] ~~Distributivity: ~~~ For all~ a, b, c:
\vspace{-2mm}\begin{center}
$a \sqcup (b \sqcap c) = (a \sqcup b) \sqcap (a \sqcup c)$\\
$a \sqcap (b \sqcup c) = (a \sqcap b) \sqcup (a \sqcap
c)$
```

```
\end{center}

\item [{BA4}] \ \ Neutral elements: ~~~ For all $a$: \vspace{-2mm}
\begin{center}
$a \sqcup \hat{0} = a$~~~~~~ $a \sqcap \hat{1} = a$
\end{center}

\item [{BA5}] ~~``Complement": ~~~ For all $a$: \vspace{-2mm}
\begin{center}
$a \sqcup  a^\prime = \hat{1}$~~~~~~ $a \sqcap  a^\prime =
\hat{0}$
\end{center}
\end{itemize}
\newpage
Prover9 input is as follows:
 {\footnotesize \begin{verbatim}
    formulas(assumptions).

    (all a all b (a | b <-> b | a)).
    (all a all b (a & b <-> b & a)).
    (all a all b all c ((a | b) | c <-> a | (b | c))).
    (all a all b all c ((a & b) & c <-> a & (b & c))).
    (all a all b all c (a | (b & c) <-> (a | b) & (a | c))).
    (all a all b all c (a & (b | c) <-> (a & b) | (a & c))).
    (all a (a | $F <-> a)).
    (all a (a & $T <-> a)).
    (all a (a | -a <-> $T)).
    (all a (a & -a <-> $F)).
    end_of_list.

    formulas(goals).

    (all a (a | a <-> a)).

    end_of_list.
\end{verbatim} }

Prover9 output is as follows:
 {\footnotesize \begin{verbatim}
    % -------- Comments from original proof --------
    % Proof 1 at 0.00 (+ 0.03) seconds.
    % Length of proof is 2.
    % Level of proof is 1.
    % Maximum clause weight is 0.
    % Given clauses 0.

    11 (all a (a | a <-> a)) # label(non_clause) # label(goal).  [goal].
    12 $F.  [deny(11)].
\end{verbatim} }

\noindent
\ This proof is fairly straightforward (outlined fully below).
\begin{itemize}
\item $a \sqcup \hat{0} = a$
```

We start with a definition from our neutral element (<u>BA4</u>).
\item $a \sqcup (a \sqcap a^\prime) = a$

The complement rule (<u>BA5</u>) allows us to make a substitution.
\item $(a \sqcup a) \sqcap (a \sqcup a^\prime) = a$

The <u>distributivity</u> (<u>BA3</u>) of our algebra allows for this substitution.
\item $(a \sqcup a) \sqcap \hat{1} = a$

The complement rule (<u>BA5</u>) is again used to generate (1) or (TRUE).
\item $(a \sqcup a)= a$

Finally, the fact that (1) is neutral (<u>BA4</u>) allows it to be ignored when paired
with (AND) or (UNION).
\end{itemize}

\newpage
\section*{Appendix: Input/Output Files}
1.INPUT:

 {\footnotesize \begin{verbatim}
    set(<u>ignore_option_dependencies</u>). % GUI handles dependencies

if(<u>Prover9</u>). % Options for <u>Prover9</u>
  assign(<u>max_weight</u>, 25).
  set(<u>restrict_denials</u>).
  assign(<u>new_constants</u>, 1).
  assign(<u>max_seconds</u>, 60).
<u>end_if</u>.

if(<u>Mace4</u>).   % Options for <u>Mace4</u>
  assign(<u>max_seconds</u>, 60).
<u>end_if</u>.

formulas(assumptions).

(a & -c) -> b.
(d | e | f) <-> c.
(h & g) <-> f.
a.
-d.
-e.
-h.
-g.

<u>end_of_list</u>.

formulas(goals).

b.

<u>end_of_list</u>.
\end{verbatim} }

1.OUTPUT:

```
 {\footnotesize \begin{verbatim}
============================== prooftrans ==============================
Prover9 (32) version Dec-2007, Dec 2007.
Process 232496 was started by Khalil on Khalil-PC,
Thu Feb  2 01:33:39 2017
The command was "/cygdrive/c/Program Files (x86)/Prover9-Mace4/bin-win32/
prover9".
============================== end of head ==============================

============================== end of input ==============================

============================== PROOF ==============================

% -------- Comments from original proof --------
% Proof 1 at 0.01 (+ 0.00) seconds.
% Length of proof is 15.
% Level of proof is 4.
% Maximum clause weight is 4.
% Given clauses 0.

1 a & -c -> b # label(non_clause).  [assumption].
2 d | e | f <-> c # label(non_clause).  [assumption].
3 h & g <-> f # label(non_clause).  [assumption].
4 b # label(non_clause) # label(goal).  [goal].
5 -a | c | b.  [clausify(1)].
9 d | e | f | -c.  [clausify(2)].
11 h | -f.  [clausify(3)].
13 a.  [assumption].
14 -d.  [assumption].
15 -e.  [assumption].
16 -h.  [assumption].
18 -b.  [deny(4)].
19 c.  [back_unit_del(5),unit_del(a,13),unit_del(c,18)].
20 f.  [back_unit_del(9),unit_del(a,14),unit_del(b,15),unit_del(d,19)].
21 $F.  [back_unit_del(11),unit_del(a,16),unit_del(b,20)].

============================== end of proof ==============================

\end{verbatim} }
```

2.INPUT:

```
 {\footnotesize \begin{verbatim}
    set(ignore_option_dependencies). % GUI handles dependencies

    if(Prover9). % Options for Prover9
      assign(max_weight, 25).
      set(restrict_denials).
      assign(new_constants, 1).
      assign(max_seconds, 60).
    end_if.

    if(Mace4).   % Options for Mace4
```

```
      assign(max_seconds, 60).
    end_if.

    formulas(assumptions).

    (all a all b (a | b <-> b | a)).
    (all a all b (a & b <-> b & a)).

    (all a all b all c ((a | b) | c <-> a | (b | c))).
    (all a all b all c ((a & b) & c <-> a & (b & c))).

    (all a all b all c (a | (b & c) <-> (a | b) & (a | c))).
    (all a all b all c (a & (b | c) <-> (a & b) | (a & c))).

    (all a (a | $F <-> a)).
    (all a (a & $T <-> a)).

    (all a (a | -a <-> $T)).
    (all a (a & -a <-> $F)).

    end_of_list.

    formulas(goals).

    (all a (a | a <-> a)).

    end_of_list.
\end{verbatim} }
```

2.OUTPUT:

```
 {\footnotesize \begin{verbatim}
============================= prooftrans =============================
Prover9 (32) version Dec-2007, Dec 2007.
Process 234724 was started by Khalil on Khalil-PC,
Thu Feb  2 02:16:52 2017
The command was "/cygdrive/c/Program Files (x86)/Prover9-Mace4/bin-win32/
prover9".
============================= end of head =============================

============================= end of input =============================

============================= PROOF =============================

% -------- Comments from original proof --------
% Proof 1 at 0.01 (+ 0.00) seconds.
% Length of proof is 2.
% Level of proof is 1.
% Maximum clause weight is 0.
% Given clauses 0.

11 (all a (a | a <-> a)) # label(non_clause) # label(goal).  [goal].
12 $F.  [deny(11)].

============================= end of proof =============================
```

```
\end{verbatim} }

\end{document}
```