Assignment 1 (*Marks: 7, Due date: Sept. 12*) **COL703, Aug-Nov 2022**

You are required to submit answers of only those questions that carry marks.

1. For the sequents below, show which ones are valid and which ones aren't:
   (a) $\neg p \to \neg q \vdash q \to p$
   (b) **[0.5 marks]** $\neg p, p \vee q \vdash q$
   (c) $p \vee q, \neg q \vee r \vdash p \vee r$
   (d) **[0.5 marks]** $p \wedge \neg p \vdash \neg(r \to q) \wedge (r \to q)$
2. Show that a formula ϕ is valid iff $\top \equiv \phi$, where $\top$ is an abbreviation for an instance $p \vee \neg p$ of LEM.
3. **[1 marks]** Let us introduce a new connective $\phi \leftrightarrow \psi$ which should abbreviate $(\phi \to \psi) \wedge (\psi \to \phi)$. Design introduction and elimination rules for $\leftrightarrow$ and show that they are derived rules if $\phi \leftrightarrow \psi$ is interpreted as $(\phi \to \psi) \wedge (\psi \to \phi)$.
4. Prove the validity of the following sequents:
   (a) $\vdash \neg p \to (p \to (p \to q))$
   (b) $p \wedge q \vdash \neg(\neg p \vee \neg q)$
   (c) **[0.5 marks]** $(p \to r) \wedge (q \to r) \vdash p \wedge q \to r$
   (d) **[0.5 marks]** $p \to q \wedge r \vdash (p \to q) \wedge (p \to r)$
5. Write a python program that takes two inputs: i) a *formula* file, containing a CNF formula in DIMACS format, and ii) a *proof* file that contains a resolution proof, and prints "correct" if the resolution proof is correct, and "incorrect" otherwise. The *proof* file contains a line for each application of the resolution rule, in the following format:

   $\texttt{ip jf l}_{i_1}\ \texttt{l}_{i_2}\ \ldots\ \texttt{l}_{i_k}\ \texttt{0}$

   which means that the clause in line i of the *proof* file (denoted by the letter 'p' after the number i) and j of the *formula* file (denoted by the letter 'f' after the number j) have been resolved to get a clause with the literals $l_{i_1}, l_{i_2}, \ldots, l_{i_k}$ (for example, `3p 7f 2 -4 5 0` denotes that the clause in line 3 of the *proof* file and line 7 of the *formula* file have been resolved, and the resulting clause is $(p_2 \vee \neg p_4 \vee p_5)$ assuming p_i is the name of the i^{th} variable).

   You are required to submit the following:

   (a) **[1.5 marks]** Your code, along with a `README` file containing instructions run the code.
   (b) **[0.5 marks]** A *formula* file that has at least 5 different clauses, and three proof files – two correct, and one incorrect – such that each proof file has at least four lines, and no two lines in any single file resolve the same two clauses.

   Here's a sample *formula*:

```
c CNF formula (p1 ∨ !p2) ∧ (p2 ∨ p3) ∧ (!p1 ∨ !p2 ∨ p3) ∧ (!p3)
p cnf 3 4
1 -2 0
2 3 0
-1 -2 3 0
-3 0
```

And, here is a sample *proof*:

```
3f 4f 1 3 0
4f 5f -1 3 0
1p 2p 3 0
6f 3p 0
```

Note that this proof is correct (we have seen this example in the class), and your code should print "correct" for the above inputs.

6. Find a propositional logic formula ϕ which contains only the atoms p, q, and r, and which is true only when p and q are false, or when $\neg q \wedge (p \vee r)$ is true.

7. **[1 marks]** An adequate set of connectives for propositional logic is a set such that for every formula of propositional logic there is an equivalent formula with only connectives from that set. For example, $\{\neg, \vee\}$ is adequate. Is $\{\leftrightarrow, \neg\}$ adequate. Justify your answer. Recall that $\phi \leftrightarrow \psi$ is interpreted as $(\phi \rightarrow \psi) \wedge (\psi \rightarrow \phi)$.

8. Show that the following sequents are not valid by finding a valuation in which the truth values of the formulas to the left of $\vdash$ are `T` and the truth value of the formula to the right of $\vdash$ is `F`.

   (a) $\neg p \vee (q \rightarrow p) \vdash \neg p \wedge q$

   (b) **[0.5 marks]** $\neg r \rightarrow (p \vee q), r \wedge \neg q \vdash r \rightarrow q$

   (c) $p \rightarrow (\neg q \vee r), \neg r \vdash \neg q \rightarrow \neg p$

   (d) **[0.5 marks]** $p \rightarrow (q \rightarrow r) \vdash p \rightarrow (r \rightarrow q)$

9. Give a natural deduction proof of PBC using only basic rules.