Exercise Sheet 1, Part I

September 29, 2022

Exercise 1. The operator $\uparrow$, also known as Sheffer stroke or NAND gate, has the following truth table:

$\uparrow$	0	1
0	1	1
1	1	0

Show that any propositional formula with language $\{\wedge, \vee, \rightarrow, \neg, 0, 1\}$ can be equivalently expressed using only the language $\{\uparrow\}$

Exercise 2. Recall the "Decision and Simplification" proof system (Infer$_D$):

$$\frac{F \quad G}{F\,[x := 0] \vee G\,[x := 1]}\,\text{CA} \qquad \frac{F}{F'}\,\textsc{Simp} \text{ with } F \rightsquigarrow F'$$

where $\rightsquigarrow$ is the result of applying a finite sequence of any of the following simplifications anywhere in the formula:

$$\begin{gathered} 0 \wedge F \rightsquigarrow 0 \qquad F \wedge 0 \rightsquigarrow 0 \qquad 1 \wedge F \rightsquigarrow F \qquad F \wedge 1 \rightsquigarrow F \\ 0 \vee F \rightsquigarrow F \qquad F \vee 0 \rightsquigarrow F \qquad 1 \vee F \rightsquigarrow 1 \qquad F \vee 1 \rightsquigarrow 1 \\ \neg 0 \rightsquigarrow 1 \qquad \neg 1 \rightsquigarrow 0 \end{gathered}$$

For each of the following formulas, express it using only the connectors $\neg$, $\wedge$ and $\vee$. Then prove they are tautologies by assuming their inverse and deriving 0 using Infer$_D$.

1. $((a \rightarrow b) \rightarrow a) \rightarrow a$
2. $\neg(a \wedge b) \rightarrow (\neg a \vee \neg b)$
3. $((\neg a \rightarrow b) \wedge (a \rightarrow b)) \rightarrow b$

Exercise 3.

1. Consider the two following small programs:

```
def f1(a:Boolean, b:Boolean, c:Boolean): Boolean = {
  if a || b then
    if b && !a then
      b && c
    else
      !b && c
  else
    c
}
```

```
def f2(a:Boolean, b:Boolean, c:Boolean): Boolean = {
  if c then
    if a then
      !b
    else true
  else false
} ensuring (_ == f1(a, b, c))
```

   Are they equivalent on all inputs? Show that they are or aren't by expressing those programs as propositional formulas. Note that the ensuring clause is automatically proved or disproved by Stainless!

2. Can any formula be expressed with only **if** a **then** b **else** c, where a, b and c are recursively constructed the same way? What if a is restricted to be a variable?

Exercise 4. For each of the following propositional logic formulas determine whether it is valid or not. If it is valid, prove it (using any acceptable mathematical argument of your choice), otherwise give a counterexample. (From *The Calculus of Computation* by A.R. Bradley and Z. Manna.)

1. $(P \land Q) \to P \to Q$
2. $(P \to Q) \lor (P \land \neg Q)$
3. $(P \to Q \to R) \to P \to R$
4. $(P \to Q \lor R) \to P \to R$
5. $\neg(P \lor Q) \to R \to \neg R \to Q$
6. $(P \to Q) \to P \to Q$
7. $((P \to Q) \to P) \to P$

8. $((P \to Q) \to P) \to Q$

9. $(\neg Q \to \neg P) \to P \to Q$

10. $(\neg R \to \neg Q \to \neg P) \to P \to Q \to R$

11. $(P \lor Q) \to (P \lor \neg Q) \to (\neg P \lor \neg Q) \to P$

12. $(P \lor Q) \to (P \lor \neg Q) \to (\neg P \lor \neg Q) \to Q$

13. $\neg(P \land Q) \to P \to \neg Q$