COMP4161 S2/2018
Advanced Topics in Software Verification

Assignment 1

This assignment starts on Mon, 2018-08-06 and is due on Mon, 2018-08-13, 23:59h. We will accept plain text (.txt) files, PDF (.pdf) files, and Isabelle theory (.thy) files.

The assignment is take-home. This does NOT mean you can work in groups. Each submission is personal. For more information, see the plagiarism policy: https://student.unsw.edu.au/plagiarism

Submit using `give` on a CSE machine:

```
give cs4161 a1 files ...
```

For example:

```
give cs4161 a1 a1.thy a1.pdf
```

1 Types (25 marks)

1. Construct a type derivation tree for the term $\lambda a\ b\ c.\ a\ (b\ c)\ (x\ c\ c)$.

   Each node of the tree should correspond to the application of a *single* typing rule, indicating which typing rule is used at each step.

   Under which contexts is the term type correct? (12 marks)

2. Find a term that has type $('a \Rightarrow 'b) \Rightarrow ('b \Rightarrow 'c) \Rightarrow 'a \Rightarrow 'c$.
   Give a type derivation tree. (10 marks)

3. Find terms s and t such that s β-reduces to t, s is ill-typed (i.e., is not well-typed), and t is well-typed. (3 marks)

2 λ-Calculus (20 marks)

Recall the encoding of booleans and booleans operations in lambda calculus seen in the lecture:

$$\begin{array}{lcl} \texttt{true} & \equiv & \lambda x\ y.\ x \\ \texttt{false} & \equiv & \lambda x\ y.\ y \\ \texttt{if} & \equiv & \lambda z\ x\ y.\ z\ x\ y \\ \texttt{or} & \equiv & \lambda x\ y.\ \texttt{if}\ x\ \texttt{true}\ y \end{array}$$

(a) Show that the β normal form for `or false true` is `true`. Justify your answer by providing the β reduction steps leading from the term to its normal form. Each step should only reduce *one* redex (i.e. one reduction per step). Ideally, you would underline the redex being reduced. (10 marks)

(b) Provide a type for `true`. Justify your answer by providing a derivation tree. (5 marks)

(c) What is a type of `or false true`? Justify your answer. (5 marks)

3 Higher-Order Unification (10 marks)

Find a unifier (substitution) for the schematic variables in the following term so that its left- and right-hand sides are $\alpha\beta\eta$-equivalent. Justify your answer by showing that the two sides $\alpha\beta\eta$-reduce to the same term.
$(\lambda y\ x.\ ?H\ x\ y) =_{\alpha\beta\eta} (\lambda x\ y.\ ?G\ (y\ x))$ (10 marks)

4 Propositional Logic (45 marks)

Prove each of the following statements, using only the proof methods `rule`, `erule`, `assumption`, and `cases`; and using only the proof rules `impI`, `impE`, `conjI`, `conjE`, `disjI1`, `disjI2`, `disjE`, `notI`, `notE`, `iffI`, `iffE`, `iffD1`, `iffD2`, `ccontr`, `classical`, `FalseE`, `TrueI`, `conjunct1`, `conjunct2`, and `mp`. You do not need to use all of these methods and rules.

Do not use `cases`, `ccontr`, `classical` for (f) nor for (j).

(a) $A \longrightarrow A \vee B$ (2 marks)

(b) $A \wedge B \longrightarrow A$ (2 marks)

(c) $(P \vee P) = P$ (3 marks)

(d) $\neg \neg P \longrightarrow P$ (3 marks)

(e) $P \longrightarrow \neg \neg P$ (3 marks)

(f) $\neg \neg \neg P \longrightarrow \neg P$ (4 marks)

(g) $(A \wedge B \longrightarrow C) = (A \longrightarrow B \longrightarrow C)$ (5 marks)

(h) $(x = \mathit{False}) = (\neg\ x)$ (5 marks)

(i) $(P \longrightarrow Q) = (\neg\ (P \wedge \neg\ Q))$ (5 marks)

(j) $P \vee \neg P \longrightarrow \neg \neg P \longrightarrow P$ (5 marks)

(k) $P \vee Q \wedge R \longrightarrow (P \vee Q) \wedge (P \vee R)$ (6 marks)

List the statements above that are provable only in a classical logic. (2 marks)