

COMP 4161 S2/08

Advanced Topics in Software Verification

Assignment 1

This assignment starts on Wed, 13.8.2008 and is due on Wed, 20.8.2008, 13:30h. Please submit electronically by email to gerwin.klein@nicta.com.au. Accepted formats are plain text, Isabelle theory files and pdf documents.

1 β Reduction (10 marks)

Write down the steps for reducing the following terms to β normal form

(a) $(\lambda x. y (\lambda v. x v)) (\lambda y. v y)$

(b) $(\lambda n. \lambda f x. f (n f x)) ((\lambda n. \lambda f x. f (n f x)) (\lambda f x. x))$

2 Encodings (15 marks)

Given the encoding of logic in untyped λ calculus from the lecture, provide a similar encoding for functions fs , sn , and $pair$ such that $fs (pair a b) =_{\beta} a$ and $sn (pair a b) =_{\beta} b$

Provide both the definition of the functions as well as β reductions that show their given characteristic properties.

3 Types (15 marks)

Construct a type derivation tree for $\lambda x y z. x y (y z)$

4 Unification (10 marks)

Find a unifier (substitution) such that $\lambda x y. ?F x = \lambda x y. c (?G y x)$

5 Proofs in Propositional Logic (50 marks)

Prove the following lemmas in Isabelle. You may use only the rules `notI`, `notE`, `conjI`, `conjE`, `disjI1`, `disjI2`, `disjCI`, `disjE`, `impI`, `impE`, `iffI`, `iffE`, and `classical` in single step rule applications with the proof methods `rule`, `erule`, `assumption`, and `case_tac`.

(a) $((A \vee B) \vee C) \longrightarrow A \vee (B \vee C)$ (6 marks)

(b) $(A \vee A) = (A \wedge A)$ (6 marks)

- (c) $(A \longrightarrow B \longrightarrow C) = (A \wedge B \longrightarrow C)$ (6 marks)
- (d) $(A \longrightarrow B \longrightarrow C) \longrightarrow (A \longrightarrow B) \longrightarrow A \longrightarrow C$ (6 marks)
- (e) $(A \longrightarrow \neg B) = (B \longrightarrow \neg A)$ (6 marks)
- (f) $((A \wedge \neg B) \vee (B \wedge \neg A)) = (A = (\neg B))$ (10 marks)
- (g) $\neg(A \wedge B) \longrightarrow \neg A \vee \neg B$ (10 marks)

For question 5, please submit an Isabelle theory file that contains all required lemmas and is processed without errors by Isabelle 2008. You can (but do not have to) use the template file provided on the lecture home page. End partial/incomplete proofs with the command `oops`.