McMaster University
Department of Computing and Software
Musa Al-hassy

COMP SCI 3EA3
Quiz 2
Solution Hints

COMP SCI 3EA3 — Software Specification and Correctness

January 30, 2017

Name _____ Student Number _____

Special Instructions:

- This examination paper includes 5 pages (including this cover page) and 6 questions.
  You are responsible for ensuring that your copy of the paper is complete. Bring any discrepancy to the attention of your invigilator.
- This is a **closed book** examination.
  No books, notes, texts, calculator or academic aids of any kind are permitted.
- **Read each question completely and carefully** before answering it.
- Answer all questions.
- **In doubt, document!**

All the questions are closely related to or directly from the lectures and sheets!

Contents

1 How to express oneself? — 6 marks —

1. Give a grammar for *terms*.

   Solution Hints:

   A term is a constant c, a varaible named x, or a function symbol f applied to other terms.

   $$t ::= c \,|\, x \,|\, f(t_1, \ldots, t_n)$$

   $$\mathsf{Term} ::= \mathsf{constant} \,|\, \mathsf{variable} \,|\, \text{application to other terms}$$

   This problem is from the January 18 lecture slides.

2. Give the inductive principle for the type of extended integers,

   $$\mathsf{Maybe}_{\mathbb{Z}} ::= \mathbb{Z} \,|\, \bot$$

   —you need not worry about type coercion—

   Solution Hints:
   The induction principle is: for any predicate $P : \mathsf{Maybe}_{\mathbb{Z}} \to \mathbb{B}$,

   $$(\forall s : \mathbb{Z} \bullet P\ s) \land P\ \bot \implies (\forall t : \mathsf{Maybe}_{\mathbb{Z}} \bullet P\ t)$$

   This problem is a question on Sheet 3.

2 ACSL Formalisation — 9 marks —

1. Using

```
/*@ axiomatic Context
{
  logic real join(real x, real y);
  logic real meet(real x, real y);
}
*/
```

   as lattice operations over the reals, *specify* the *Golden Rule* in ACSL notation.

   Solution Hints:

```
//@ lemma GR: \forall real x, y;    meet(x,y) == x  <==>  join(x,y) == y;
```

   The class was explicitly instructed to be able to do this along with hint "this may be on the quiz" ; January 25.

2. Specify the axioms needed to make `(real , op , e)` into a *monoid* in ACSL notation.

   Solution Hints:

   Recall that a monoid is an set endowed with an associative operation and an identity element:

```
/*@ axiomatic MonoidAxioms
{
  logic real e;
  logic real op(real x, real y);

  axiom opAxiom: \forall real x, y, z; op(x, op(y, z)) == op(op(x,y), z);
  axiom eAxiom:  \forall real x, y, z; op(x, e) == x  && op(e, x) == x;
}
*/
```

   This problem is a question on Sheet 2.

3 Erroneous Specfication — 8 marks —

Given the problem

"write a program to calculate the mean of two numbers x and y"

one proposed specfication is

$$\{\ true\ \}\ ?\ \{\ mean = (x + y)/2\ \}$$

1. As *succinctly* as possible, explain why this is a *bad* specification.

   Solution Hints:
   It says nothing about the original values of the givens; in-particular, we can satisfy the specification by assigning all variables involved to zero!

2. In the Hoare-triple format, give a correct specification conveying the intended behaviour.

   Solution Hints:
   A more accurate *definition* would be,

$$\{\ x = X\ \wedge\ y = Y\ \}\ ?\ \{\ mean = (x + y)/2\ \wedge\ x = X\ \wedge\ y = Y\ \}$$

   The very first day we discussed how having no assertions on variables means we can just set them to a constant such as 0 and life gets easy. We even did an involved example on sorting an array; and many participated.

4 Half The Quadrivia — 8 marks —

1. State the "Leibniz rule".

   Solution Hints:
   It states that equal things can be substituted for equal things,

   $$x = y \Rightarrow f\ x = f\ y$$

   More formally known as the "identity of indiscernibles, it states that "entities are indiscernible precisely when they share all their properties in common:"

   $$x = y \ \equiv\ (\forall P \bullet P\ x = P\ y)$$

   The class was explicitly told to learn this specific rule and it was hinted that it may be on the quiz ; January 20.

2. Using the "duality principle for lattices" form the dual of the statement $x \sqcap y \sqsubseteq x$ and *simplify* as much as possible.

   Solution Hints:

   $$\begin{array}{ll} & \mathsf{dual}\,(x \sqcap y \sqsubseteq x) \\ = & \quad \{\ \text{definition, } \mathsf{dual}\ S = S[(\sqsubseteq, \sqcap, \sqcup) := (\sqsupseteq, \sqcup, \sqcap)], \\ & \qquad \text{and textual substitution}\ \} \\ & x \sqcup y \sqsupseteq x \\ = & \quad \{\ \text{definition, } r \sqsupseteq l \ \equiv\ l \sqsubseteq r\ \} \\ & x \sqsubseteq x \sqcup y \end{array}$$

   Recall that the initial statement has the name "weakening" while the statement at the end of the calculation has the name "strengthening". That is, weakening and strengthening are dual results!

   Not only was the class informed this problem was on the quiz, this particular example was done in class ; January 27.

5 Mental Execution — 8 marks —

Give precondition and postcondition formulae for the following program.

$$r, x := 1, 0;\ \textbf{do}\ x \neq B \rightarrow r := r \cdot A; x := x + 1\ \textbf{od}$$

Solution Hints:
The loop multiplies r by A repeatedly and then increments x, and as such it simply assigns r to be some exponent of A. In particular, the loop terminates when $x = B$ and so this program sets $r = A^B$. Indeed this is similar to the "Puzzle" on Sheet 3.

Since the loop *increments* x each time and terminates only when $x = B$, we need $x \leq B$ as part of the program invariant. In particular, we need this to hold initially; that is, when $x = 0$ and so we necessarily need $0 \leq B$ for the program to terminate.

With these two in hand, we have

$$\{\ 0 \leq B\ \}\ r, x := 1, 0;\ \textbf{do}\ x \neq B \rightarrow r := x \cdot A; x := x + 1\ \textbf{od}\ \{\ r = A^B\ \}$$

Similar to the 'Puzzle' of Sheet 3 but in GCL rendition. No trick intended; just rewarding those who tried the sheet.

6 Complete The Proof — 10 marks —

All boxes are worth 1 mark, while the first box of part 2 is worth 4 marks.

Fill in the blank boxes with the appropriate *names* of theorems or properties.

Solution Hints:

1. When the order of arguments does not matter for an operation, we say it is *commutative*; for Boolean-valued operations the synonym *symmetric* is used. This property was an exercise on Sheet 2. Moreover, the Golden Rule appeared in a previous question on this quiz.

   Proving $p \wedge q \equiv q \wedge p$**:**

   $p \wedge q$
   $=$ { Golden Rule }
   $p \equiv q \equiv p \vee q$
   $=$ { Symmetry of $\equiv$, which says $p \equiv q \equiv q \equiv p$ }
   $q \equiv p \equiv p \vee q$
   $=$ { Symmetry of $\vee$, which says $p \vee q \equiv q \vee p$ }
   $q \equiv p \equiv q \vee p$
   $=$ { Golden Rule , again }
   $q \wedge p$

   First and last boxes are Golden Rule, which I told them to know, and placed in question 2. Other boxes are symmetry, from Sheet 2.

2. In an arbitrary lattice $(L, \sqsubseteq, \sqcap, \sqcup)$,

   Proving $x \sqcup x = x$**:**

   $x \sqcup x$
   $=$ { The principle of "indirect equality from above" says two items are identical precicely when they share identical ancestors, ie $l = r \equiv (\forall a \mid a \in L \bullet l \sqsubseteq a \equiv r \sqsubseteq a)$
   So for any a in our lattice, we have
   $x \sqcup x \sqsubseteq a$
   $=$ { join-characterisation }
   $x \sqsubseteq a \wedge x \sqsubseteq a$
   $=$ { Idempotencey of $\wedge$ }
   $x \sqsubseteq a$
   hence $x \sqcup x = x$. }
   x

   This exact example was done in class ; January 27. The class was instructed to be able to reproduce the proof since it may be on the quiz.