Exam Software Specification, Verification and Testing, October 24, 2018

The exam duration is 02:30:00 (two hours and 30 minutes). For your convenience, please remember to use the Exam24102018.hs file available on Canvas.

Problem 1 (10p) Assume that x ranges over integer numbers, while n is a natural number. State your answers for the following questions and describe how you found out.

1. (2p) Consider the following correct Hoare statement
   $\{...\}\{\lambda x \mapsto x^2 + 2x\}\{\lambda x \mapsto x > 0\}$.
   Which valid precondition is weaker: $\lambda x \mapsto x > -2$ **or** $\lambda x \mapsto x > 0$?

2. (2p) Consider the following correct Hoare statement
   $\{\lambda x \mapsto x > 0\}\{\lambda x \mapsto x^2 - Ax\}\{...\}$.
   Assuming A is an integer number, which valid postcondition is stronger: $\lambda x \mapsto x > 5$ **or** $\lambda x \mapsto x > A$?

3. (2p) Consider the Fibonacci recursion $F_0 = 0, F_1 = 1, F_{n+2} = F_{n+1} + F_n, n \geq 0$.
   What is the weakest valid precondition for the Hoare statement
   $\{...\}\{\lambda n \mapsto F_{n+1}F_{n-1} - F_n^2\}\{\lambda n \mapsto n > 0\}$?

4. (2p) Consider the Fibonacci recursion $F_0 = 0, F_1 = 1, F_{n+2} = F_{n+1} + F_n, n \geq 0$.
   What is the strongest valid postcondition for the Hoare statement
   $\{\lambda n \mapsto n > 5\}\{\lambda n \mapsto F_{n+1}F_{n-1} - F_n^2\}\{...\}$?

Problem 2 (15p)

1. (4p) Consider the set $A = \{0, 1, 2\}$. Give all equivalences that have no singleton subsets, that is $\forall x \in A\ \exists y \in A, x \neq y$, such that xRy. Express these relations as sets of pairs and as partitions.

2. (4p) How many equivalences without singletons are on a set with n elements? Prove your findings by induction.

3. (7p) On a set A of cardinality n, consider the relations $R = \{(x, y) | \pi(x) = y\} \cup \Delta_A$, $\Delta_A = \{(a, a) | a \in A\}$, where $\pi(i)$ is a permutation on A. How many such relations are equivalences? State your answer and describe how you found out.

Problem 3 (10p) Let the relation R on the set $A = \{0, 1, 2, 3\}$ be given by

$$R = \{(0, 2), (1, 3), (2, 0)\}.$$

1. (2p) Determine R^+, the transitive closure of R (write it as a list of pairs).

2. (4p) Let $S = R^+ \cup \Delta_A$, $\Delta_A = \{(a,a)|a \in A\}$. Is S an equivalence on A? State your answer and describe how you found out.

3. (4p) Prove by induction that the fixed point of the function `aClos s = s ∘ s ∪ s` represents the transitive closure of a relation `s`.

Problem 4 (15p) A relation R is serial on a domain A if for any $x \in A$ there is an $y \in A$ such that xRy. Suppose relations are represented as lists of pairs:

```
type Rel a = [(a,a)]
```

1. (5p) Write a function for checking whether a relation is serial:

   ```
   isSerial :: Eq a => [a] -> Rel a -> Bool
   isSerial = error "not yet implemented"
   ```

2. (5p) Test your implementation with two QuickCheck properties.

3. (5p) Consider the relation $R = \{(x,y)|x = y(\texttt{mod n})\}$, where `(mod n)` is the modulo function in modular arithmetic and $n > 0$. Discuss whether (and when) R is serial. How can you test whether R is serial? How can you prove that R is serial?

Problem 5 (10p) Consider the following implementation of a function `sub::Form->Set Form` that finds all the sub-formulae of a given formula.

```
module Exam
    where
import SetOrd
type Name = Int

data Form = Prop Name
          | Neg  Form
          | Cnj [Form]
          | Dsj [Form]
          | Impl Form Form
          | Equiv Form Form
          deriving (Eq,Ord)

sub :: Form -> Set Form
sub (Prop x) = Set [Prop x]
sub (Neg f) = unionSet (Set [Neg f]) (sub f)
sub f@(Cnj [f1,f2]) = unionSet ( unionSet (Set [f]) (sub f1)) (sub f2)
sub f@(Dsj [f1,f2]) = unionSet ( unionSet (Set [f]) (sub f1)) (sub f2)
sub f@(Impl f1 f2) = unionSet ( unionSet (Set [f]) (sub f1)) (sub f2)
sub f@(Equiv f1 f2) = unionSet ( unionSet (Set [f]) (sub f1)) (sub f2)
```

1. (5p) How can you prove that the `sub` implementation is correct? Test the imple-

mentation with two QuickCheck properties.

2. (5p) Write a recursive implementation of the function `nsub :: Form -> Int` such that `nsub f` computes the exact number of sub-formulae of the formula `f`. Test your implementation using Quickcheck.

Problem 6 (20p) Remember that we can write any number N$\geq$ 60 as 6X+11Y, with X,Y integers ≥ 0.

```
data Tree a = T a [Tree a] deriving (Eq,Ord,Show)

grow :: (node -> [node]) -> node -> Tree node
grow step seed = T seed (map (grow step) (step seed))

tree n = grow (step611 n) (6,11)
```

1. (10p) Implement a `step611` function for a tree `tree n` such that every natural number $60 \leq M \leq n$ occurs in `tree n`.

2. (10p) Can you check that your implementation is correct? Describe how and what your findings are.

Problem 7 (20p) Consider the following investment problem (**Gordon Growth Model**): An investor plans on buying company shares. The current price per share is d_0 and the annual rate of return is g, while the market interest rate is r. We also assume that $g \leq r$, i.e. the return rate may not exceed the interest.

The investor calculates the present (discounted) value of his investment over the next n years, as follows: he sums up the discounted value of his dividend after k-years, for $k = 1$ to n; in formula, he/she calculates

$$P_0 = \sum_{k=1}^{n} \frac{d_0(1+g)^k}{(1+r)^k} = d_0 \sum_{k=1}^{n} \left(\frac{1+g}{1+r}\right)^k .$$

You are asked to investigate the following four implementations of the present (discounted) value:

```
gordon1 d0 n g r = d0*(1-((1+g)/(1+r))^n)/(1-(1+g)/(1+r))

gordon2 d0 n g r = d0*sum [((1+g)/(1+r))^k|k <-[1..n]]

gordon3 d0 n g r = d0*(1-((1+g)/(1+r))^n)/(r-g)*(1+g)

gordon4 d0 n g r = sum [d0*((1+g)/(1+r))^k|k <-[0..n-1]]
```

Which of these implementations is correct? Or are they all flawed? Or maybe they are all correct? State your answer and describe how you found out.