SFWR ENG 2F04 Assignment 3: More Propositional PVS & Predicate Logic Syntax & Interpretations

Due: 1720 Tuesday October 22, 2002

1. Understanding PVS (25 marks total) In a PVS file, write down a PVS theorem called `C1` that you would attempt to prove to demonstrate that the following argument is valid:

   $$p \rightarrow q, \neg(q \wedge \neg r), p \models r$$

   Since syntax and semantics agree in propositional logic, we can show

   $$p \rightarrow q, \neg(q \wedge \neg r), p \vdash r$$

   obtain our conclusion by the soundness of our proof system.

   a) Invoke the PVS prover on the theorem and apply the (FLATTEN) command. You should obtain the sequent:

   ```
   {-1}    (p IMPLIES q)
   {-2}    p
     |-------
   {1}     (q & NOT r)
   {2}     r

   Rule?
   ```

   Recall that PVS effectively does the proofs in reverse. Thus PVS is saying that to show

   $$p \rightarrow q, \neg(q \wedge \neg r), p \vdash r \tag{1}$$

   begin by showing

   $$p \rightarrow q, p \vdash (q \wedge \neg r) \vee r \tag{2}$$

   Suppose that on line n of your proof you have shown (2). Show what the next lines of a formal proof would be to use this to show (1) on line $n + k$.

   b) The PVS command (SPLIT -1) is applied to the first equation in the premises of the sequent in (b). What propositional proof rules is PVS effectively using in reverse to do this splitting? What sequents result and why is one of them trivially true? Finish the proof using whatever PVS commands you desire.

   c) Do a formal proof of the valid argument by hand i.e., show

   $$p \rightarrow q, \neg(q \wedge \neg r), p \models r$$

   by showing that

   $$p \rightarrow q, \neg(q \wedge \neg r), p \vdash r$$

Continued on page 2

2. **Predicate Logic Formulas as Specifications and Programs as Models:**

   Consider the following set of formulas:

$$\begin{aligned}\Gamma \;:=\; & \{\forall x \neg Q(x, f(x)),\\ & \forall x \forall y (Q(f(x), f(y)) \rightarrow Q(x, y)),\\ & \exists x \forall y \neg Q(x, f(y))\}\end{aligned}$$

   The intended interpretation is given by a structure of the form $\mathcal{M} := \langle A, Q^{\mathcal{M}}, f^{\mathcal{M}} \rangle$ where:

$$\begin{aligned} A &:= \mathbb{N} = \{0, 1, 2, \ldots\}\\ Q^{\mathcal{M}} &:= \{(x, y) \in \mathbb{N} | x = y\}\end{aligned}$$

   and the interpretation of f, denoted $f^{\mathcal{M}}$ is yet to be determined.

   In your summer job as a software developer[1] you are given Γ as a formal specification and told to create two different versions of $f^{\mathcal{M}}$.

   a) First, find an $f^{\mathcal{M}}$ such that for this interpretation of f we have $\mathcal{M} \not\models \Gamma$.

   b) Next, find an $f^{\mathcal{M}}$ such that for this interpretation of f we have $\mathcal{M} \models \Gamma$.

3. Huth+Ryan p. 101 1, 3, 5, 6, 7(a)(i)(k)

4. Huth+Ryan p. 108 1, 2

5. Huth+Ryan p. 135 1, 3, 4, 6

6. Huth+Ryan p. 139 1, 2, 5, 6

The End

[1] You aren't "real" software engineers until we get accredited, then you graduate, get the required experience, and the pass the Law & Ethics exams.