Advanced Programming Languages
Homework Assignment 5F

EECS 590

Logistics. You must work alone. Your name and Michigan email address must appear on the first page of your PDF submission but *may not appear anywhere else*. This is to protect your identity during peer review. The first page of your submission is *not* shared during peer view but all subsequent pages are.

Exercise 5F-1. Bookkeeping [2 points]. These answers should appear on the first page of your submission and are kept private.

1. Indicate in a sentence or two how much time you spent on this homework.

2. Indicate in a sentence or two how difficult you found it subjectively.

All subsequent answers should appear after the first page of your submission and may be shared publicly during peer review.

Exercise 5F-2. VCGen Do-While [8 points]. Choose exactly *one* of the two options below. (If you are not certain, pick the first. The answers end up being equivalent, but the first may be easier to grasp for some students and the second easier to grasp for others.)

- Give the (backward) verification condition formula for the command $\mathsf{do}_{Inv}\ c$ while b with respect to a post-condition P. The invariant *Inv* is true before each evaluation of the predicate b. Your answer may not be defined in terms of VC(while...).

- Give the (backward) verification condition formula for the command $\mathsf{do}_{Inv1,Inv2}\ c$ while b with respect to a post-condition P. The invariant *Inv1* is true before c is first executed. The invariant *Inv2* is true before each evaluation of the loop predicate b. Your answer may not be defined in terms of VC(while...).

Exercise 5F-3. VCGen Mistakes [20 points]. Consider the following three alternate while Hoare rules (named lannister, stark, and targaryen):

$$\frac{\vdash \{X\}\; c\; \{b \implies X\; \wedge\; \neg b \implies Y\}}{\vdash \{b \implies X\; \wedge\; \neg b \implies Y\}\; \mathsf{while}\; b\; \mathsf{do}\; c\; \{Y\}}\; \mathsf{lannister} \qquad \frac{\vdash \{X\; \wedge\; b\}\; c\; \{X\}}{\vdash \{X\}\; \mathsf{while}\; b\; \mathsf{do}\; c\; \{X\}}\; \mathsf{stark}$$

$$\frac{\vdash \{X\}\; c\; \{X\}}{\vdash \{X\}\; \mathsf{while}\; b\; \mathsf{do}\; c\; \{X\; \wedge\; \neg b\}}\; \mathsf{targaryen}$$

All three rules are sound but incomplete. Choose **two** incomplete rules. For each chosen rule provide the following:

1. the name of the rule and

2. A and

3. B and

4. σ and

5. σ' and

6. c such that

7. $\langle c, \sigma \rangle \Downarrow \sigma'$ and

8. $\sigma \models A$ and

9. $\sigma' \models B$ but

10. it is not possible to prove $\vdash \{A\}\; c\; \{B\}$.

Flavor text: Incompleteness in an axiomatic semantics or type system is typically not as dire as unsoundness. An incomplete system cannot prove all possible properties or handle all possible programs. Many research results that claim to work for the C language, for example, are actually incomplete because they do not address `setjmp`/`longjmp` or bitfields. (Many of them are also unsound because they do not correctly model unsafe casts, pointer arithmetic, or integer overflow.)

Submission. Turn in the formal component of the assignment as a single PDF document via the `gradescope` website. Your name and Michigan email address must appear on the first page of your PDF submission but may not appear anywhere else.