Programming Languages [Fall 2017]
Test III

NAME: _____

Instructions:

1) This test is 9 pages in length.

2) You have 2 hours to complete and turn in this test.

3) This test is closed books, notes, laptops, phones, smartwatches, friends, neighbors, etc.

4) Use the backs of pages in this test packet for scratch work. If you write more than a final answer in the area next to a question, circle your final answer.

5) Write and sign the following: "I pledge my Honor that I have not cheated, and will not cheat, on this test."

_____

_____

Signed: _____

1. [45 points]
Consider a CBV λ_{ST} having bool as the base type, but then with six extra features added: if-then-elses, returns, references, exceptions, expression-sequencing $(e_1; \ldots; e_n)$, and for-loops $for(e_1;e_2;e_3)\{e_4\}$. All have standard meanings, with left-to-right evaluation. For-loops evaluate to false upon completion.

Using the notations and conventions from class, define first-order abstract syntax and dynamic and static semantics for this language, as would be appropriate for a proof of type safety. The next page is blank, in case you need additional space.

2. [55 points]
Now consider R, a CBV λ_{ST} having unit as the base type, but then with only one extra feature added: references. Evaluation is left to right and *not* defined with evaluation contexts. The following subproblems ask you to state and/or prove properties of R. Note that properties of R may be stated a bit differently than for simple PLs lacking references; e.g., the Weakening Lemma is: $(\Lambda_1 \subseteq \Lambda_2 \ \wedge \ \Gamma_1 \subseteq \Gamma_2 \ \wedge \ \Lambda_1,\Gamma_1 \vdash e:\tau) \Rightarrow (\Lambda_2,\Gamma_2 \vdash e:\tau)$.

(a) State, but don't prove, the Canonical Forms Lemma.

(b) Prove the Substitution Lemma for variable expressions in R.
(Hint: For λ_{ST}, Substitution says: $(\Gamma \cup \{x:\tau\} \vdash e':\tau' \wedge \Gamma \vdash e:\tau) \Rightarrow \Gamma \vdash [e/x]e':\tau'$)

(c) State the Progress Theorem, and prove that it holds for the cases of location (l) and dereferencing (!e) expressions. Assume, without proof, that all the standard lemmas (Weakening, Inv., CF, and Subs.) hold. If your proof ever uses a nonstandard lemma, state and prove that lemma. The next page is blank, in case you need additional space.

(d) State the Preservation Theorem, and prove that it holds for the cases of reference-introducing (ref e) expressions. Assume, without proof, that all the standard lemmas (Weakening, Inv., CF, and Subs.) hold. If your proof ever uses a nonstandard lemma, state and prove that lemma. The next page is blank, in case you need additional space.

[Undergraduates stop here. The remaining problem is for graduate students.]

(e) [10 points]
State and prove type safety, assuming that Progress and Preservation have already been proved.