CIS 5000: Software Foundations **Quiz 8** Nov 20, 2025

Name (printed): ______________________ Penn ID (8-digit #): ____________

Please write your name as it is displayed on Gradescope. Indicate your answers with a cross mark ☒. To correct a mistake, fill in the incorrect box (like this: ■) and make a cross in the correct box.

$\boxed{1}$ Is the following term stuck?

```
iszero (if true then (succ 0) else 0)
```

(a) ☐ Yes

☐ No

(b) If not, what rule or rules are used to derive the next evaluation step:

$\boxed{2}$ Is the following term stuck?

```
succ (if true then true else true)
```

(a) ☐ Yes

☐ No

(b) If not, what rule or rules are used to derive the next evaluation step:

3 Suppose we add the following two new rules to the reduction relation:

```
| ST_PredTrue :
  pred true --> pred false
| ST_PredFalse :
  pred false --> pred true
```

Does this preserve...

- ☐ Determinism of the step relation?
- ☐ Preservation?
- ☐ Progress?
- ☐ Termination?

4 Recall the definition of progress:

```
Theorem progress : forall t T,
  <{ |-- t \in T }> ->
  value t \/ exists t', t --> t'.
```

As well as the definition of strong progress from `Smallstep`:

```
Theorem strong_progress : forall t,
  value t \/ exists t', t --> t'.
```

What is the difference between these two theorems?

- ☐ There is no difference; they are equivalent.
- ☐ Progress implies strong progress.
- ☐ Strong progress implies progress.
- ☐ They have no relationship.
- ☐ Other

5 In the language defined in `Types`...

(a) Every well-typed normal form is a value.

☐ True ☐ False

(b) Every value is a normal form.

☐ True ☐ False

(c) The single-step reduction relation is a partial function (i.e., it is deterministic).

☐ True ☐ False

(d) The single-step reduction relation is a total function.

☐ True ☐ False

For Reference

```
        -------------------------------                    (ST_IfTrue)
        if true then t1 else t2 --> t1

        -------------------------------                    (ST_IfFalse)
        if false then t1 else t2 --> t2

                    t1 --> t1'
------------------------------------------------           (ST_If)
if t1 then t2 else t3 --> if t1' then t2 else t3

                  t1 --> t1'
              --------------------                         (ST_Succ)
              succ t1 --> succ t1'

                ------------                               (ST_Pred0)
                pred 0 --> 0

              numeric value v
             -------------------                           (ST_PredSucc)
             pred (succ v) --> v

                 t1 --> t1'
              --------------------                         (ST_Pred)
              pred t1 --> pred t1'

               -----------------                           (ST_IsZero0)
               iszero 0 --> true

              numeric value v
           -------------------------                       (ST_IsZeroSucc)
           iszero (succ v) --> false

                t1 --> t1'
            ------------------------                       (ST_IsZero)
            iszero t1 --> iszero t1'
```