CIS 5000: Software Foundations Quiz 2 September 11, 2025

Name (printed): _____ Penn ID (8-digit #): _____

- Please write your name as it is displayed on Gradescope.
- Indicate your answers with a cross mark ☒. If you need to correct an answer, fill in the incorrect box ■ and add a cross in the correct box ☒.

1 Recall the definition on binary numbers:

```
Inductive bin : Type :=
  | Z
  | B0 (n : bin)
  | B1 (n : bin).
```

(a) We would like to implement a function which doubles a binary number.

```
Definition double (b: bin) : bin   (* := ... *)
```

We would like it to be correct...

```
Lemma double_correctness: forall b: bin,
  bin_to_nat (double b) = 2 * bin_to_nat b.
```

and ideally, it should also preserve normalized numbers (see Reference page for definitions):

```
Lemma double_normalization: forall b: bin,
  normalize b = b ->
  normalize (double b) = double b.
```

Which properties are satisfied by the following implementations?

- ```
  Definition double1 (b: bin): bin :=
    B0 b.
  ```

  ☐ Correctness and normalization ☒ Correctness only ☐ Neither

- ```
  Definition double2 (b: bin): bin :=
    match b with
    | Z => Z
    | _ => B0 b
    end.
  ```

  ☒ Correctness and normalization ☐ Correctness only ☐ Neither

(b) Complete the following implementation of a function adding two binary numbers by filling in values for the "unknowns" (marked with names beginning `?h`) below. The `incr` function is defined on the Reference handout.

```
Fixpoint add_bin (m n: bin): bin :=
  match m with
  | Z => n
  | B0 m' =>  match n with
              | Z => m
              | B0 n' => ?h1 (add_bin m' n')
              | B1 n' => ?h3 (add_bin m' n')
              end
  | B1 m' =>  match n with
              | Z => m
              | B0 n' => ?h5 (add_bin m' n')
              | B1 n' => ?h6 (?h7 (add_bin m' n'))
              end
  end.
```

?h1 is... ☒ B0 ☐ B1 ☐ incr

?h2 is... ☐ B0 ☒ B1 ☐ incr

?h3 is... ☐ B0 ☒ B1 ☐ incr

?h4 is... ☒ B0 ☐ B1 ☐ incr

?h5 is... ☐ B0 ☐ B1 ☒ incr

?h6 is... ☒ B0 ☐ B1 ☐ incr

?h7 is... ☐ B0 ☐ B1 ☒ incr

2 Which of the following functions can be implemented in Rocq?

- `Fixpoint nth {X: Type} (ls: list X) (n: nat): X`

  ☐ Yes ☒ No

- `Fixpoint nth_default {X: Type} (ls: list X) (n: nat) (x: X): X`

  ☒ Yes ☐ No

- `Fixpoint nth_error {X: Type} (ls: list X) (n: nat): option X`

  ☒ Yes ☐ No

Reference

```
Definition double_bin (b:bin) : bin :=
  match b with
  | Z => Z
  | _ => B0 b
  end.

Fixpoint incr (m:bin) : bin :=
  match m with
  | Z     => B1 Z
  | B0 m' => B1 m'
  | B1 m' => B0 (incr m')
  end.

Fixpoint normalize (b:bin) : bin :=
  match b with
  | Z    => Z
  | B0 b' => double_bin (normalize b')
  | B1 b' => incr (double_bin (normalize b'))
  end.
```

Normalize can be implemented in various ways. The important thing to understand is that `normalize` transforms any binary number into a normalized number—that is, a number with no leading zeroes—without changing its value.