1 Induction

Let’s inductively define a set of integers Quux with the following inference rules.

**Rule 1** \( 8 \in \text{Quux} \)

**Rule 2** \( 5 \in \text{Quux} \)

**Rule 3** \( a \in \text{Quux}, b \in \text{Quux} \implies c = a + b + 1 \)

(a) Of the rules above (i.e., Rule 1, Rule 2, and Rule 3), which are axioms and which are inductive rules?

(b) Give a derivation showing that 11 is in the set Quux.

(c) Give a derivation showing that 20 is in the set Quux.

(d) Write down the inductive reasoning principle for Quux. That is, if you wanted to prove that for some property \( P \), for all \( a \in \text{Quux} \) we have \( P(a) \), what would you need to show? (See Lecture 3 §2.2 and §2.3.)

(e) Prove that for all \( a \in \text{Quux} \), there exists \( i \in \mathbb{Z} \) such that \( a = 3 \times i - 1 \).

Make sure that you follow the Recipe for Inductive Proofs! See Lecture 3 §2.5. What set are you inducting on? What is the property you are trying to prove? Go through each case.

(f) Is 2 in the set Quux? If so, give a derivation proving it.

2 Small-step operational semantics

Consider the small-step operational semantics for the language of arithmetic expressions (Lecture 2). Let \( \sigma_0 \) be a store that maps all program variables to zero.

(a) Show a derivation that \( \langle 3 + (5 \times \text{bar}), \sigma_0 \rangle \longrightarrow \langle 3 + (5 \times 0), \sigma_0 \rangle \).

(b) What is the sequence of configurations that \( \langle \text{foo} := 5; (\text{foo} + 2) \times 7, \sigma_0 \rangle \) steps to? (You don’t need to show the derivations for each step, just show what configuration \( \langle \text{foo} := 5; (\text{foo} + 2) \times 7, \sigma_0 \rangle \) steps to in one step, then two steps, then three steps, and so on, until you reach a final configuration.)

(c) Find an integer \( n \) and store \( \sigma' \) such that \( \langle ((6 + (\text{foo} := (\text{bar} := 3; 5); 1 + \text{bar})) + \text{bar}) \times \text{foo}, \sigma_0 \rangle \longrightarrow^* \langle n, \sigma' \rangle \).

(d) Is the relation \( \longrightarrow \) reflexive? Is it symmetric? Is it anti-symmetric? Is it transitive?

(For each of these questions, if the answer is “no”, what is a suitable counterexample? If any of the answers are “yes”, think about how you would prove it.)

3 Large-step operational semantics

Consider the large-step operational semantics for the language of arithmetic expressions (Lecture 4). Let \( \sigma_0 \) be a store that maps all program variables to zero.

(a) Show a derivation that \( \langle 3 + (5 \times \text{bar}), \sigma_0 \rangle \Downarrow \langle 3, \sigma_0 \rangle \).
(b) Find an integer $n$ and store $\sigma'$ such that $\langle \text{foo} := 5; (\text{foo} + 2) \times 7, \sigma_0 \rangle \Downarrow (n, \sigma')$.

  If you have time and a big piece of paper, give the derivation of $\langle \text{foo} := 5; (\text{foo} + 2) \times 7, \sigma_0 \rangle \Downarrow (n, \sigma')$.

(c) Is the relation $\Downarrow$ reflexive? Is it symmetric? Is it anti-symmetric? Is it transitive?

  (For each of these questions, if the answer is “no”, what is a suitable counterexample? If any of the answers are “yes”, think about how you would prove it.)