Correctness of a Simple Compiler

Daniel Britten, The University of Waikato

November 30, 2020

Based upon the example discussed by Professor Graham Hutton on Computerphile on November 27: https://youtu.be/T_IINWzQhow

Proof based upon Associate Professor Adam Chlipala's proof: http://adam.chlipala.net/cpdt/html/StackMachine.html

1 Implementation

```
Inductive Expr :=
  | Val (n : Z)
  | Add (e1 e2 : Expr).

Inductive Op :=
  | PUSH (n : Z)
  | ADD.

Fixpoint eval (e : Expr) :=
  match e with
  | Val n ⇒ n
  | Add x y ⇒ eval x + eval y
  end.

Fixpoint comp (e : Expr) :=
  match e with
  | Val n ⇒ [PUSH n]
  | Add x y ⇒ comp x ++ comp y ++ [ADD]
  end.

Fixpoint exec (ops : list Op) (stack : list Z) : list Z :=
  match ops, stack with
  | PUSH n :: c, s ⇒ exec c (n :: s)
  | ADD :: c, (m::n::s) ⇒ exec c (n + m :: s)
  | _, s ⇒ s
  end.
```

2 Examples

```
Example eg1 : exec (comp (Add (Val 42) (Val 42))) [] = [84]. reflexivity. Qed.
Example eg2 : eval (Add (Val 42) (Val 42)) = 84. reflexivity. Qed.
Example eg3 : exec ([PUSH 1; PUSH 2; PUSH 3]) [] = [3;2;1]. reflexivity. Qed.
Example eg4 : exec ([ADD; PUSH 1; PUSH 2; PUSH 3]) [] = []. reflexivity. Qed.
```

3 Correctness Proof

First we need to strengthening the induction hypothesis as described in: Adam Chlipala's similar proof at http://adam.chlipala.net/cpdt/html/StackMachine.html

```
Lemma correct_helper :
  ∀ e ops s, exec (comp e ++ ops) s = exec ops (eval e :: s).
  Proof.
    induction e.
     - simpl. reflexivity.
     - intros. simpl.
       rewrite app_assoc_reverse.
       rewrite IHe1.
       rewrite app_assoc_reverse.
       rewrite IHe2.
       simpl. reflexivity.
  Qed.
```

Now the proof follows from the lemma.

```
Theorem correct : ∀ e, exec (comp e) [] = [eval e].
  Proof.
    intros.
    pose (correct_helper e [] []) as H.
    rewrite app_nil_r in H.
    assumption.
  Qed.
```

QED! - We now have the highest degree of certainty (proof) that our implementation meets the specification: ∀ e, *exec* (*comp* e) [] = [eval e]

The Coq source file of this proof is available at: https://coda-coda.github.io/blog/program_correctness_based_on_computerphile/coq_simple_compilation_correctness.v