

Oct 20, 16 7:09

ImpStep.v

Page 1/2

```

Require Import List.
Require Import String.
Require Import ZArith.

Open Scope list_scope.
Open Scope string_scope.
Open Scope Z_scope.

Require Import ImpSyntax.
Require Import ImpCommon.
Require Import ImpEval.

Inductive step :
  store -> stmt ->
  store -> stmt -> Prop :=
| step_set :
  forall s x e v,
  eval_e s e v ->
  step
  s (Sset x e)
  (update s x v) Snop
| step_ifelse_t :
  forall s e p1 p2,
  eval_e s e (Vbool true) ->
  step
  s (Sifelse e p1 p2)
  s p1
| step_ifelse_f :
  forall s e p1 p2,
  eval_e s e (Vbool false) ->
  step
  s (Sifelse e p1 p2)
  s p2
| step_while_t :
  forall s e p,
  eval_e s e (Vbool true) ->
  step
  s (Swhile e p)
  s (Sseq p (Swhile e p))
| step_while_f :
  forall s e p,
  eval_e s e (Vbool false) ->
  step
  s (Swhile e p)
  s Snop
| step_seq_nop :
  forall s p2,
  step
  s (Sseq Snop p2)
  s p2
| step_seq :
  forall s p1 p2 s' p1',
  step
  s p1
  s' p1' ->
  step
  s (Sseq p1 p2)
  s' (Sseq p1' p2).

Inductive step_star :
  store -> stmt ->
  store -> stmt -> Prop :=
| step_star_refl :
  forall s p,
  step_star
  s p
  s p
| step_star_l :
  forall s1 p1 s2 p2 s3 p3,

```

Oct 20, 16 7:09

ImpStep.v

Page 2/2

```

  step
  s1 p1
  s2 p2 ->
  step_star
  s2 p2
  s3 p3 ->
  step_star
  s1 p1
  s3 p3.

```