

Oct 24, 16 7:17

ImpSyntax.v

Page 1/1

```
Require Import String.
Require Import ZArith.
```

```
Inductive val : Type :=
| Vbool : bool -> val
| Vint  : Z -> val
| Vstr  : string -> val
| Vaddr : Z -> val.
```

```
Inductive op1 : Type :=
| Oneg
| Onot.
```

```
Inductive op2 : Type :=
| Oadd
| Osub
| Omul
| Odiv
| Omod
| Oeq
| Olt
| Ole
| Oconj
| Odisj.
```

```
Inductive expr : Type :=
| Eval : val -> expr
| Evar : string -> expr
| Eop1  : op1 -> expr -> expr
| Eop2  : op2 -> expr -> expr -> expr
| Elen  : expr -> expr
| Eidx  : expr -> expr -> expr.
```

```
Definition store : Type :=
  list (string * val).
```

```
Definition heap : Type :=
  list val.
```

```
Inductive stmt : Type :=
| Snop : stmt
| Sset : string -> expr -> stmt
| Salloc : string -> expr -> expr -> stmt
| Swrite : string -> expr -> expr -> stmt
| Scall : string -> string -> list expr -> stmt
| Sifelse : expr -> stmt -> stmt -> stmt
| Swhile : expr -> stmt -> stmt
| Sseq : stmt -> stmt -> stmt
| Sincall : stmt -> expr -> string -> store -> stmt.
```

```
Inductive func : Type :=
| Func : string -> list string -> stmt -> expr -> func.
```

```
Inductive prog : Type :=
| Prog : list func -> stmt -> expr -> prog.
```