

Languages-beta: SL-Funcons-Index *

The PPlanCompS Project

SL-Funcons-Index.cbs | PLAIN | PRETTY

OUTLINE

Computations

- Normal computation
 - Flowing
 - Giving
 - Binding
 - Generating
 - Storing
 - Interacting
 - Input
 - Output
- Abnormal computation
 - Terminating abruptly
 - Failing
 - Returning
 - Breaking
 - Continuing

Values

- Value Types
- Primitive values
 - Booleans
 - Integers
 - The null value
- Composite values
 - Lists
 - Strings
 - Maps
 - Objects
- Abstraction values
 - Generic abstractions
 - Functions

*Suggestions for improvement: plancomps@gmail.com.
Reports of issues: <https://github.com/plancomps/CBS-beta/issues>.

Computations

Normal computation

Flowing

- [*Funcon* sequential
- Alias* seq
- Funcon* effect
- Funcon* if-true-else
- Alias* if-else
- Funcon* while-true
- Alias* while]

Giving

- [*Funcon* initialise-giving
- Funcon* give
- Funcon* given]

Binding

- [*Type* environments
- Alias* envs
- Datatype* identifiers
- Alias* ids
- Funcon* initialise-binding
- Funcon* bind-value
- Alias* bind
- Funcon* bound-value
- Alias* bound
- Funcon* closed
- Funcon* scope
- Funcon* collateral]

Generating

- [*Funcon* fresh-atom]

Storing

- [*Funcon* initialise-storing
- Datatype* variables
- Alias* vars
- Funcon* allocate-initialised-variable
- Alias* alloc-init
- Funcon* assign
- Funcon* assigned]

Interacting

Input

- [*Funcon* read]

Output

```
[ Funcon  print ]
```

Abnormal computation

Terminating abruptly

```
[ Funcon  finalise-abrupting ]
```

Failing

```
[ Funcon  fail  
  Funcon  else  
  Funcon  checked ]
```

Returning

```
[ Funcon  return  
  Funcon  handle-return ]
```

Breaking

```
[ Funcon  break  
  Funcon  handle-break ]
```

Continuing

```
[ Funcon  continue  
  Funcon  handle-continue ]
```

Values

Value Types

```
[ Type  values  
  Alias  vals  
  Type  cast-to-type  
  Alias  cast  
  Funcon  is-equal  
  Alias  is-eq ]
```

Primitive values

Booleans

```
[ Datatype  booleans  
  Alias  bools  
  Funcon  true  
  Funcon  false  
  Funcon  not ]
```

Integers

```
[ Type    integers
  Alias   ints
  Funcon  integer-add
    Alias  int-add
  Funcon  integer-subtract
    Alias  int-sub
  Funcon  integer-multiply
    Alias  int-mul
  Funcon  integer-divide
    Alias  int-div
  Funcon  integer-negate
    Alias  int-neg
  Funcon  integer-is-less
    Alias  is-less
  Funcon  integer-is-less-or-equal
    Alias  is-less-or-equal
  Funcon  integer-is-greater
    Alias  is-greater
  Funcon  integer-is-greater-or-equal
    Alias  is-greater-or-equal
  Funcon  decimal-natural
    Alias  decimal ]
```

The null value

```
[ Datatype  null-type
  Funcon    null-value
    Alias    null ]
```

Composite values

Lists

```
[ Datatype  lists
  Funcon    list-nil
    Alias    nil
  Funcon    list-cons
    Alias    cons
  Funcon    list-head
    Alias    head
  Funcon    list-tail
    Alias    tail ]
```

Strings

```
[ Type    strings
  Funcon  string-append
  Funcon  to-string ]
```

Maps

```
[ Funcon  map  
  Funcon  map-lookup  
    Alias  lookup  
  Funcon  map-override ]
```

Objects

```
[ Datatype  objects  
  Funcon  object  
  Funcon  object-feature-map ]
```

Abstraction values

Generic abstractions

```
[ Funcon  closure ]
```

Functions

```
[ Datatype  functions  
  Funcon  function  
  Funcon  apply ]
```