Haskell Haskell "" Haskell """" Haskell HaskellMLCleanCoqAgda

Haskell

 Haskell Scheme #fScheme "" #tScheme "" Scheme "" Scheme Lisp Lisp `nil`"" `nil` ""

Scheme Lisp Scheme `'()` "" #f `bool`Lisp C C 0 "" 0 "" C

```
int i = 0;
...
...
if (i++) { ...}
```

Scheme `nil` "" #f ""Scheme

```
(or x y z)
```

 #f #t

```
(cond
 [(or x y z)
  => (lambda (found)
       (do-something-with found))])
```

```
(let ([found (first-non-false x y z)])
  (cond
   [(not (eq? found #f))
    (do-something-with found)]))
```

 Scheme ""`=> (lambda (found) ...)` `(or x y z)` `found` #f "" `(or x y z)` `cond` `found` `cond`

 `let` #f "" Scheme ""

Haskell Haskell Haskell category theory Haskell

 Haskell category theory "abstract nonsense""" category theory

 Haskell "" Haskell"" Haskell Scheme Java C++ shell Haskell Haskell Haskell Haskell Haskell ""
monad Haskell

 Haskell Haskell

 C++ C++ C++""

 C++ C++ C++ """"——""

 C++ functor functor C++ first-class function ""C++ functor Scheme lambda functor class
class functor field functor "" field Scheme lambda

 functor C++ functor functor """"C++ functor C++ 1983 Scheme 1975 Lisp 1958 C++ Scheme
8 Scheme lexical scoping lambdafunctor lambda C++ boost

20111111C++ Bjarne Stroustrup Indiana C++11 Andrew boost Stroustrup 11

 Stroustrup C++11 Scheme "" C++ ""

 C++

 """"""""——