

## EVAL記法

- LISPプログラムに直結する書き方

( + 1 6 ) ⇒ 7

( oddp ( + 1 6 ) ) ⇒ t

( \* 3 ( + 1 6 ) ) ⇒ 21

- 内側の括弧から評価する

25

## EVALルール1

- 数, 記号T, 記号NILはそれ自身が値となる

例 23 ⇒ 23

t ⇒ t

nil ⇒ nil

26

## EVALルール2

- リストの評価  
最初の要素が関数名を表す  
2つ目以降の要素はそれの引数  
引数は左から右の順序で評価する

例 (oddp (+ 1 6))

27

## 関数の定義

例 (defun average (x y)  
(/ (+ x y) 2.0))

上を評価すると関数averageが使えるよう  
になり, (average 3 6) ⇒ 4.5

x と y は局所変数となる

28

## EVALルール3

- シンボルを評価すると, その名前の変数の値となる  
外部変数の例 `pi`  $\Rightarrow$  3.14159
- シンボルそれ自身を値としたい時は, シンボルの前に引用符(アポストロフィ)をつける  
例 `'saito` `'blue`  
(注意 `t` と `nil` には不要)  
`'A` は (QUOTE A) と同値

29

## EVALルール4

- 引用符がついたオブジェクトを評価すると, 引用符がつかない値となる

例 `'tokyo`  $\Rightarrow$  `tokyo`  
`'(+ 2 3)`  $\Rightarrow$  `(+ 2 3)`  
`'(+ 1 (+ 2 3))`  $\Rightarrow$  `(+ 1 (+ 2 3))`  
`(equal '(a b c) (a b c))`  $\Rightarrow$  ???  
`(third (my name is saito))`  $\Rightarrow$  ???

30

# 演習プリント1