Making decisions

Umut Özge

COGS 502: Symbols and Programming
METU, Informatics

Special symbols

```
* 17
17
* A
→ error   A is unbound
* t
T
* nil
NIL
*
```

Predicates

```
* (numberp 3)
T
* (numberp +)
NIL
*
```

unary predicates

zerop
evenp
integerp
oddp
⋮

binary predicates

(= 3 4) ⟹ NIL
(< 3 4) ⟹ T
(<= 3 4) ⟹ T

IF

(if (+ 3 4) (* 8 7) (- 4 9)) ⇒ 56

(+ 3 4) ⇓ 7

(* 8 7) ⇓ 56

returned

(if (< 3 4) 8 7)

(< 3 4) ⇓ T

⇒ 8

evaluate and return the value.

(if <test> <form-for-success> <form-for-failure>)

<test> → NIL

<test> → sth not NIL → <form-for-success>

<form-for-failure>

evaluate and return the value.

A simple procedure

- Take an integer n;
  - return $n/2$, if n is even;
  - return $3n + 1$, otherwise.

4 => 2

3 => 10

COND

- Take an integer n;
  - return $n/2$, if n is even;
  - return $3n + 1$, if n is divisible by 3;
  - return n itself, otherwise.

```
(if (evenp n)
    (/ n 2)
    (+ 1 (* 3 n)))

                                  clause
(cond ((evenp n) (/ n 2))
      ((oddp n) (+ 1 (* 3 n))))
        +
(cond ((evenp n) (/ n 2))
      (T (+ 1 (* 3 n))))
```

(and <form-1> <form-2> ... <form n>)

⇓ non-nil ⇓ non-nil ⇓ value → return value of the 'and' form.

y

(and (zerop 0) (> 3 2) (+ 4 5))

10

(and (* 2 3) (+ 3 7))

⇓

6

(or <f1> <f2> ... <fn>)

nil ⇓

return value of 'or' form. 8