

Tema 9: Funciones

El constructor *deffunction*

Introducción

- El constructor *deffunction* permite crear nuevas funciones dentro de CLIPS.
- Componentes de *deffunction*:
 - Un nombre.
 - Un comentario (opcional).
 - Una lista de cero o más parámetros regulares.
 - Un parámetro comodín opcional que gestiona un número variable de argumentos.
 - Una secuencia de acciones o expresiones, que se ejecutarán de forma secuencial cuando se llame a la función definida mediante este constructor.

Introducción

- Valor devuelto:
 - El valor devuelto es el resultado de evaluar la última acción.
 - Si la función no tiene acciones, devuelve el símbolo FALSE.
 - Si ocurre un error durante la ejecución de la función, se abortan las acciones que no se hayan completado y la función devuelve el símbolo FALSE.
- Una función sólo puede ser llamada si ha sido previamente definida (salvo en el caso de funciones recursivas).

Sintaxis

```
(defunction <nombre> [<comentario>]
  (<parámetro-regular>* [<parámetro-comodín>])
  <acción>*)
```

<parámetro-regular> ::= <variable-monocampo>

<parámetro-comodín> ::= <variable-multicampo>

Ejemplo

```
CLIPS> (deffunction print-args (?a ?b $?c)
  (printout t ?a " " ?b " y " (length$ $?c)
    " extras: " $?c crlf))
CLIPS> (print-args 1 2)
1 2 y 0 extras: ()
CLIPS> (print-args a b c d)
a b y 2 extras: (c d)
```

Ejemplo

```
CLIPS> (deffunction factorial (?a)
  (if (or (not (integerp ?a)) (< ?a 0))
    then (printout t "Error factorial." crlf)
    else
      (if (= ?a 0)
        then 1
        else (* ?a (factorial (- ?a 1))))))

CLIPS> (factorial Hola)
Error factorial.

CLIPS> (factorial 4)
24
```