

Abstract Syntax Tree

Linus Szillat

June 6, 2022

In this manual we cover the functionality of BSL-Tools.

Contents

| | | |
|----------|------------------------------------|----------|
| 1 | BSL Tree Tests | 4 |
| 1.1 | Constant Definition | 4 |
| 1.2 | Struct Definition | 4 |
| 1.3 | Function Definition | 4 |
| 1.4 | Conditional Expression | 4 |
| 1.5 | Function Call Expression | 4 |
| 1.6 | Name Expression | 4 |
| 1.7 | List Expression | 4 |
| 1.8 | Value Expressions | 5 |
| 1.9 | Consecutive Program | 5 |

Here are some tests for the Show Feature of the Abstract Syntax Tree.

1 BSL Tree Tests

1.1 Constant Definition

(define one 2) (define two 3) (+ one two)

1.2 Struct Definition

(define-struct pool (people fish water)) (make-pool (2 1 a lot))

1.3 Function Definition

(define (swim-with-the-fish pool) (string-append Cool pool with a number of (number->string (pool-fish pool))))

1.4 Conditional Expression

(cond ((< 1 0) Chocolate is the best!) ((< 0 0) But Ice Cream is way better!) ((> 1000 200) Still Pizza's the best!)) (else #f))

1.5 Function Call Expression

(swim-with-the-fish (make-pool 2 20 a little less than much, but still much))

1.6 Name Expression

just-a-normal-not-intriguing-name

1.7 List Expression

(list 1 2 3 4 5)

1.8 Value Expressions

Boolean:

#t

Number:

42

String:

"Hallo Welt"

Empty-List:

empty

1.9 Consecutive Program