

Mu Operator with Expressions

Example 1 : Basic Mu (Returns the Value)

The basic mu operator finds and returns the unique value:

$$(\mu x : \mathbb{N} \mid x * x = 16)$$

This evaluates to 4 (the unique natural number whose square is 16).

Example 2 : Mu with Expression (Returns Transformed Value)

Adding a bullet and expression transforms the result:

$$(\mu x : \mathbb{N} \mid x * x = 16 \bullet x + 1)$$

This finds $x = 4$, then evaluates $x + 1$, giving 5.

Example 3 : Finding land Doubling

$$(\mu n : \mathbb{N} \mid n > 10 \wedge n < 12.2 * n)$$

Finds $n = 11$ (unique n between 10 and 12), then returns $2 * 11 = 22$.

Example 4 : Square Root land Square Again

$$(\mu x : \mathbb{N} \mid x * x = 25 \bullet x * x * x)$$

Finds $x = 5$, then returns $5^3 = 125$.

Example 5 : String Indexing Example

[String, Char]

$$\boxed{\begin{array}{l} charAt : String \times \mathbb{N} \rightarrow Char \\ findChar : String \times Char \rightarrow \mathbb{N} \end{array}}$$

$$\forall s : String \bullet \forall c : Char \bullet findChar(s, c) = (\mu i : \mathbb{N} \mid charAt(s, i) = c)$$

`findChar` returns the index of the first (unique) occurrence of character `c`.

Example 6 : With Transformation

$$\boxed{\begin{array}{l} findAndAdvance : String \times Char \rightarrow \mathbb{N} \end{array}}$$

$$\forall s : String \bullet \forall c : Char \bullet findAndAdvance(s, c) = (\mu i : \mathbb{N} \mid charAt(s, i) = c) + 1$$

Finds the index, then returns the next index.

Example 7 : Mathematical Example - Inverse Function

$$\boxed{\begin{array}{l} \text{inverse} : (\mathbb{N} \rightarrow \mathbb{N}) \times \mathbb{N} \rightarrow \mathbb{N} \\ \forall f : \mathbb{N} \rightarrow \mathbb{N} \bullet \forall y : \mathbb{N} \bullet \text{inverse}(f, y) = (\mu x : \mathbb{N} \mid f(x) = y) \end{array}}$$

For an injective function f , $\text{inverse}(f, y)$ finds the unique x such that $f(x) = y$.

Example 8 : With Computation

$$\boxed{\begin{array}{l} \text{inverseSquared} : (\mathbb{N} \rightarrow \mathbb{N}) \times \mathbb{N} \rightarrow \mathbb{N} \\ \forall f : \mathbb{N} \rightarrow \mathbb{N} \bullet \forall y : \mathbb{N} \bullet \text{inverseSquared}(f, y) = (\mu x : \mathbb{N} \mid f(x) = y \bullet x * x) \end{array}}$$

Finds the inverse, then squares it.

Example 9 : Practical - Finding Configuration Values

$[\text{ConfigKey}, \text{ConfigValue}]$

$$\boxed{\begin{array}{l} \text{configs} : \text{ConfigKey} \rightarrow\!\!\! \rightarrow \text{ConfigValue} \\ \text{getConfigInt} : \text{ConfigKey} \rightarrow \mathbb{N} \\ \text{parseValue} : \text{ConfigValue} \rightarrow \mathbb{N} \\ \\ \forall k : \text{ConfigKey} \bullet \\ k \in \text{dom configs} \Rightarrow \\ \text{getConfigInt}(k) = (\mu v : \mathbb{N} \mid \text{parseValue}(\text{configs}(k)) = v) \end{array}}$$

Finds the unique \mathbb{N} value by parsing a configuration value.

Example 10 : With Default Processing

$$\boxed{\begin{array}{l} \text{getConfigWithDefault} : \text{ConfigKey} \times \mathbb{N} \rightarrow \mathbb{N} \\ \\ \forall k : \text{ConfigKey} \bullet \forall \text{default} : \mathbb{N} \bullet k \in \text{dom configs} \Rightarrow \\ \quad \text{getConfigWithDefault}(k, \text{default}) = (\mu v : \mathbb{N} \mid \text{parseValue}(\text{configs}(k)) = v) \\ \forall k : \text{ConfigKey} \bullet \forall \text{default} : \mathbb{N} \bullet k \notin \text{dom configs} \Rightarrow \\ \quad \text{getConfigWithDefault}(k, \text{default}) = \text{default} \end{array}}$$

Uses mu to parse the value when key \exists , otherwise returns default.

Example 11 : Comparison - With and Without Bullet

Without bullet (returns the value itself):

$$\boxed{\begin{array}{l} \text{findRoot} : \mathbb{N} \\ \\ \text{findRoot} = (\mu x : \mathbb{N} \mid x * x = 49) \end{array}}$$

$\text{findRoot} = 7$

With bullet (returns a transformed value):

$$\boxed{\begin{array}{l} \text{findRootPlusOne} : \mathbb{N} \\ \\ \text{findRootPlusOne} = (\mu x : \mathbb{N} \mid x * x = 49 \bullet x + 1) \end{array}}$$

$\text{findRootPlusOne} = 8$

Example 12 : Complex Expression

$(\mu x : \mathbb{N} \mid x > 5 \wedge x < 7 \bullet (x * x) + (x + 1))$
Finds $x = 6$, then evaluates $(6 * 6) + (6 + 1) = 36 + 7 = 43$.

Example 13 : Nested Mu Expressions

$(\mu x : \mathbb{N} \mid x * x = 16 \bullet (\mu y : \mathbb{N} \mid y * y = (x + 5) \bullet y + x))$
Finds $x = 4$, then finds y where $y^2 = (4 + 5) = 9$, so $y = 3$, then returns $3 + 4 = 7$.

Example 14 : Error Conditions

The mu operator requires the constraint to have exactly one solution. If there are zero solutions or multiple solutions, mu is undefined (evaluation error).

Zero solutions:

$(\mu x : \mathbb{N} \mid x * x = -1)$ (no natural number has negative square)

Multiple solutions:

$(\mu x : \mathbb{N} \mid x < 5)$ (0, 1, 2, 3, 4 all satisfy this)

For mu to be well-defined, the constraint must uniquely identify exactly one value.

Example 15 : Design Pattern

The pattern " $(\mu x : T \mid uniqueness_constraint)$. expression" is useful when:

1. You need to find a unique value satisfying a constraint
2. You want to transform or compute something based on that value
3. The transformation is more than just returning the value itself

Common use cases:

- Database lookups with post-processing
- Finding and transforming configuration values
- Inverse functions with additional computation
- Unique element selection with modification