Background

Basics for setting up equations and extracting solutions.

- Terminologies
- Symbols
- Constants
- Operators
- ==
- Expressions
- Words
- Phrases
- Sentences

Terminologies

lhs : Left Hand Side

rhs : Right Hand Side

p = q : an identity

Equational Expression: expr1 = expr2 e.g. expr1 = a*x+b and expr2 = cos[b-c]

copy: when an object is duplicated, byte by byte so to say, into a facsimile.

> This definition of **copy** is the simplest to grasp the concept. In due places, more sophisticated definitions will be produced.

Structure: a **structure** is that which permitting **Similarity** to other **Similars**.

Example: **Circular** is a structure since it permits **similarity to** anything that is **round**.

Example: **Red** is a structure since it permits **similarity to** anything that is **red**.

Example: **Numeral** is a structure since it permits **similarity to** anything that is a **number**.

ff calls a list e.g. **{1,2,3}** as a **structure**, or a function e.g. **cos[ ]** a **structure** or an **algebraic structure** e.g. **x^2-1**.

Symbols

In Free Form Programming Language (ff) the concept of Symbol is more than one:

1. Individual alphabet in a language e.g. x in English , ω in Greek
2. Composite of individual symbols e.g. dara that while it is the concatenation of four alphabets ff treats it as a single symbol
3. Mathematical symbols e.g. ⊗
4. Glyph (Unicode) e.g. □□ , □□
5. Glyph a raster image (this option is currently unavailable for HTML use)

> The mathematical symbols in many cases have particular syntax and semantics in an expression which cannot be altered or ignored.

> The mathematical symbol single space or " " is interpreted multiplication in mathematical textbooks. Additionally in ff often " " is assigned the functionality of separator in a list of items.

o be more general ff defines a Symbol as follows:

> An atomic or indivisible structure with no discerning programmable components is called a Symbol.

Constants

In Free Form Programming Language (ff) the concept of a **Constant** is an **immutable** structure.

Examples : 5, □□ .

Operators

=

+

-

*

/

{}

()

;

""

->

<

>

<=

∩

∪

⊕

×

∂

∈

∉

⊂

→

and



==

value to rhs. What do we mean equal by value? Example: 3 = 6/2 in spite of the fact that lhs and rhs are made from very different expressions but their values are Identical or the same, hence the Identity term.

= : copies stores the rhs into lhs;

```
// = stores the rhs into lhs;
x1  = 5;

x2 = 2 * x1;

show x2;

// == creates an equations between lsh and rhs;
x3 = (x1 == 5);

show x3;


x4 = (x2 == 10);

show x4;

// lhs and rhs with different variables;
x5 = (x2 == 2*x1);

show x5;

save as equations;
```

Expressions

Algebraic

a*(c-u)

Free Form

solve a*x +b == 0 for x

Words

"I am a word"

Phrases

The highlighted are the phrases

pts = evaluate l2 in the -1 to 3 range at increments of (1/4);

Part of a sentence, have a meaning in and of itself, but often has no procedural meaning alone.

Sentences

pts = evaluate l2 in the -1 to 3 range at increments of (1/4);

> FIXME: line cannot be even be used as a subphrase;

l2 := l[v,t,{0,b}];