Def Innerproduct

= (Insert +)○(ApplyToAll ×)○Transpose

Or, in abbreviated form:

Def IP = (/+)○(αX)○Trans.

\[ IP: \langle 1,2,3, 6,5,4 \rangle = \]

Definition of IP
\[ \Rightarrow (/+)○(αX)○Trans: \langle 1,2,3, 6,5,4 \rangle \]

Effect of composition, ○
\[ \Rightarrow (/+)○(αX)○\text{Trans}^{3}\]
\[ \langle 1,2,3, 6,5,4 \rangle \]

Applying Transpose
\[ \Rightarrow (/+)○(αX): \langle 1,6,2,3, 3,4 \rangle \]

Effect of ApplyToAll, α
\[ \Rightarrow (/+)○\langle 6,10,12 \rangle \]

Applying ×
\[ \Rightarrow (+)○\langle 6,10,12 \rangle \]

Effect of Insert, /
\[ \Rightarrow (+)○\langle 6,10,12 \rangle \]

Applying +
\[ \Rightarrow (+)○\langle 6,22 \rangle \]

Applying + again
\[ \Rightarrow 28 \]

11.2 Description
An FP system comprises the following:
1) a set \( O \) of objects;
2) a set \( F \) of functions \( f \) that map objects into objects;
3) an operation, application;
4) a set \( F \) of functional forms; these are used to combine existing functions, or objects, to form new functions in \( F \);
5) a set \( D \) of definitions that define some functions in \( F \) and assign a name to each.

Selector functions
\[ l : x = x = \langle x_1, \ldots, x_n \rangle \rightarrow x_1; \perp \]

and for any positive integer \( s \)
\[ s : x = x = \langle x_1, \ldots, x_n \rangle \& n \geq s \rightarrow x_n; \perp \]

Thus, for example, 3: \( (A, B, C) = C \) and 2: \( (A) = \perp \).
Note that the function symbols 1, 2, etc. are distinct from the atoms \( l, 2 \), etc.

Tail
\[ tl : x = x = \langle x_1 \rangle \rightarrow \phi, \]
\[ x = \langle x_1, \ldots, x_n \rangle \& n \geq 2 \rightarrow \langle x_2, \ldots, x_n \rangle; \perp \]

Identity
\[ id : x = x \]
Atom
atom: \( x = \phi \) is an atom \( \rightarrow T; x \neq \bot \rightarrow F; \bot \)

Equals
\( eq \): \( x = <y, z> \ & y = z \rightarrow T; x = <y, z> \ & y \neq z \rightarrow F; \bot \)

Null
null: \( x \equiv x = \phi \rightarrow T; x \neq \bot \rightarrow F; \bot \)

Reverse
\( rev \): \( x = \phi \rightarrow \phi \);
\( x = <x_1, ..., x_n> \rightarrow <x_n, ..., x_1> \); \( \bot \)

Distribute from left; distribute from right
\( distl \): \( x = <y, \phi> \rightarrow \phi \);
\( x = <y, <z_1, ..., z_n>> \rightarrow <<y, z_1>, ..., <y, z_n>> \); \( \bot \)
\( distr \): \( x = <\phi, y> \rightarrow \phi \);
\( x = <<y_1, ..., y_n>, z> \rightarrow <<y_1, z>, ..., <y_n, z>> \); \( \bot \)

Length
length: \( x = <x_1, ..., x_n> \rightarrow n; x = \phi \rightarrow 0; \bot \)

Add, subtract, multiply, and divide
\(+: x = <y, z> \ & y, z \text{ are numbers} \rightarrow y + z; \bot \)
\(-: x = <y, z> \ & y, z \text{ are numbers} \rightarrow y - z; \bot \)
\(\times: x = <y, z> \ & y, z \text{ are numbers} \rightarrow y \times z; \bot \)
\(+: x = <y, z> \ & y, z \text{ are numbers} \rightarrow y + z; \bot \)

Transpose
\( trans \): \( x = <\phi, ..., \phi> \rightarrow \phi \);
\( x = <x_1, ..., x_n> \rightarrow <y_1, ..., y_n> \); \( \bot \)

where
\( x_i = <x_{i_1}, ..., x_{i_m}> \) and
\( y_j = <x_{j_1}, ..., x_{j_m}> \), \( 1 \leq i \leq n, 1 \leq j \leq m \).

And, or, not
\( and \): \( x = <T, T> \rightarrow T \);
\( x = <T, F> \lor x = <F, T> \lor x = <F, F> \rightarrow F; \bot \)
eTC.

Append left; append right
\( apndl \): \( x \equiv x = <y, \phi> \rightarrow <\phi> \);
\( x = <y, <z_1, ..., z_n>> \rightarrow <y, z_1, ..., z_n> \); \( \bot \)
\( apndr \): \( x \equiv x = <\phi, z> \rightarrow <z> \);
\( x = <<y_1, ..., y_n>, z> \rightarrow <y_1, ..., y_n, z> \); \( \bot \)

Right selectors; Right tail
1r: \( x = <x_1, ..., x_n> \rightarrow x_n ; \bot \)
2r: \( x \equiv x = <x_1, ..., x_n> \ & n \geq 2 \rightarrow x_{n-1}; \bot \)
eTC.

tlr: \( x \equiv x = <x_1> \rightarrow \phi \);
\( x = <x_1, ..., x_n> \ & n \geq 2 \rightarrow <x_1, ..., x_{n-1}}> ; \bot \)

Rotate left; rotate right
\( rol \): \( x = \phi \rightarrow \phi \); \( x = <x_1> \rightarrow <x_1> \);
\( x = <x_1, ..., x_n> \ & n \geq 2 \rightarrow <x_2, ..., x_n, x_1> \); \( \bot \)
eTC.
Functional Form

**Composition**

\[(f \circ g): x \equiv f:(g:x)\]

**Construction**

\[[f_1, \ldots, f_n]: x \equiv \langle f_1:x, \ldots, f_n:x \rangle\] (Recall that since \(<\ldots, \bot, \ldots> = \bot\ and\ all\ functions\ are \bot\-preserving,\ so\ is \[f_1, \ldots, f_n].\)

**Condition**

\((p \to f; g): x = (p:x) = T \to f:x; \quad (p:x) = F \to g:x; \bot\)

**Constant (Here \(x\) is an object parameter.)**

\(\tilde{x}: y = y = \bot \to \bot; x\)

**Insert**

\(\langle f:x = x = \langle x_1 \rangle \to x_1; x = \langle x_1, \ldots, x_n \rangle \& n \geq 2 \to f:x_1, f: \langle x_2, \ldots, x_n \rangle \rangle; \bot\)

If \(f\) has a unique right unit \(u_f \neq \bot\), where \(f: \langle x, u_f \rangle \in \{x, \bot\}\) for all objects \(x\), then the above definition is extended: \(f: \phi = u_f\). Thus

\[+ : \langle 4, 5, 6 \rangle = + : \langle 4, + : \langle 5, + : \langle 6 \rangle \rangle \rangle = + : \langle 4, + : \langle 5, 6 \rangle \rangle = 15\]

\[+ : \phi = 0\]

**Apply to all**

\(of: x \equiv x = \phi \to \phi; \quad x = \langle x_1, \ldots, x_n \rangle \to \langle f:x_1, \ldots, f:x_n \rangle; \bot\)

**Binary to unary (\(x\) is an object parameter)**

\((bu f x): y \equiv f: \langle x, y \rangle\)

Thus

\((bu + l): x = I + x\)

**While**

\((while pf)x = p:x = T \to (while pf):(f:x); \quad p:x = F \to x; \bot\)

**Definition**

\[\text{Def last } \equiv \text{null} + l \to 1; \text{last} + l\]
Let \( \text{last}:<1,2> \):

\begin{align*}
\text{def} \text{inition of last} & \Rightarrow (\text{null}::t \rightarrow 1; \text{last}::t):<1,2> \\
\text{def}inition of \text{form } (p \rightarrow f; g) & \Rightarrow \text{last}::t::<1,2> \\
\text{since } \text{null}::t::<1,2> & = \text{null}::<2> \\
& = F \\
\text{action of the form } f \circ g & \Rightarrow \text{last}::(t::<1,2>) \\
\text{def}inition of \text{primitive tail} & \Rightarrow \text{last}::<2> \\
\text{def}inition of \text{last} & \Rightarrow (\text{null}::t \rightarrow 1; \text{last}::t):<2> \\
\text{def}inition of \text{form } (p \rightarrow f; g) & \Rightarrow t::<2> \\
\text{since } \text{null}::t::<2> & = \text{null}::t = T \\
& = 2 \\
\text{def}inition of \text{selector } 1 & \\
\end{align*}

**Def** \( ! = \text{eq}0 \rightarrow 1; \times \circ [\text{id}, ! \circ \text{sub}1] \)

**where**

**Def** \( \text{eq}0 \equiv \text{eq}0 [\text{id}, 0] \)

**Def** \( \text{sub}1 = -\circ [\text{id}, 1] \)

Here are some of the intermediate expressions an FP system would obtain in evaluating \( !:2 \):

\( !:2 \Rightarrow (\text{eq}0 \rightarrow 1; \times \circ [\text{id}, ! \circ \text{sub}1]):2 \)

\( \Rightarrow x::\text{id}::2; ! \circ \text{sub}1::2 \Rightarrow x::<2, !::1/0> \)

\( \Rightarrow x::<2, x::<1,1>::0> \Rightarrow x::<2, x::<1,1>> \Rightarrow x::<2, 1> \Rightarrow 2. \)

11.3.2 **Inner product.** We have seen earlier how this definition works.

**Def** \( \text{IP} \equiv (+) \circ (\times) \circ \text{trans} \)

11.3.3 **Matrix multiply.** This matrix multiplication program yields the product of any pair \( <m \times n> \) of conformable matrices, where each matrix \( m \) is represented as the sequence of its rows:

\( m = <m_1, \ldots, m_r> \)

where \( m_i = <m_{i1}, \ldots, m_{ir}> \) for \( i = 1, \ldots, r. \)

**Def** \( \text{MM} = (\alpha \circ \text{IP}) \circ (\alpha \circ \text{dist} \circ [1, \text{trans} \circ 2]) \)

Laws of the algebra of programs

\([f \circ g] \circ h \equiv [f \circ h, g \circ h] \)
PROPOSITION: For all functions \( f, g, \) and \( h \) and all objects \( x, ((f \circ g) \circ h):x = [f \circ h, g \circ h]:x. \)

PROOF:

\[ ([f \circ g] \circ h):x = [f, g]:(h:x) \]

by definition of composition

\[ = \langle f,(h:x), g:(h:x) \rangle \]

by definition of construction

\[ = \langle (f \circ h):x, (g \circ h):x \rangle \]

by definition of composition

\[ = [f \circ h, g \circ h]:x \]

by definition of construction

\[ \square \]

Proofs of some law

\[ \text{apndI } [f \circ g, a \circ h] = a \circ \text{apndI } [g, h] \]

PROOF. We show that, for every object \( x, \) both of the above functions yield the same result.

CASE 1. \( h:x \) is neither a sequence nor \( \emptyset. \)

Then both sides yield \( \bot \) when applied to \( x. \)

CASE 2. \( h:x = \emptyset. \) Then

\[ \text{apndI } [f \circ g, a \circ h]:x \]

\[ = \text{apndI } \langle f \circ g:x, \emptyset \rangle = \langle f:(g:x) \rangle \]

\[ = a \circ \text{apndI } [g, h]:x \]

\[ = a \circ \langle g:x, \emptyset \rangle = a \circ \langle g:x \rangle \]

\[ = \langle f:(g:x) \rangle \]

CASE 3. \( h:x = <y_1, \ldots, y_n>. \) Then

\[ \text{apndI } [f \circ g, a \circ h]:x \]

\[ = \text{apndI } \langle f \circ g:x, a \circ f \circ <y_1, \ldots, y_n> \rangle \]

\[ = \langle f:(g:x), f:y_1, \ldots, f:y_n \rangle \]

\[ = a \circ \text{apndI } [g, h]:x \]

\[ = a \circ \langle g:x, <y_1, \ldots, y_n> \rangle \]

\[ = a \circ \langle g:x, y_1, \ldots, y_n \rangle \]

\[ = \langle f:(g:x), f:y_1, \ldots, f:y_n \rangle \]

\[ \square \]
**Proposition 2**

Pair & not-null \( \rightarrow \)

\[ \text{apndl}^\circ [[1^2, 2], \text{distr}^\circ [\text{tl} \circ 1, 2]] = \text{distr} \]

where \( f \& g \) is the function: \( \text{and}^\circ [f, g] \), and \( f^2 = f \circ f \).

**Proof.** We show that both sides produce the same result when applied to any pair \( <x, y> \), where \( x \neq \phi \), as per the stated qualification.

**Case 1.** \( x \) is an atom or \( \bot \). Then distr: \( <x, y> = \bot \), since \( x \neq \phi \). The left side also yields \( \bot \) when applied to \( <x, y> \), since \( \text{tl} \circ 1 : <x, y> = \bot \) and all functions are \( \bot \)-preserving.

**Case 2.** \( x = <x_1, \ldots, x_n> \). Then

\[
\text{apndl}^\circ [[1^2, 2], \text{distr}^\circ [\text{tl} \circ 1, 2]]: <x, y>
= \text{apndl: } <x_1, y>, \text{distr: } <\text{tl}: x, y>
= \text{apndl: } <x_1, y>, \phi = <x_1, y>> \quad \text{if } \text{tl}: x = \phi
= \text{apndl: } <x_1, y>, <x_2, y>, \ldots, <x_n, y>> \quad \text{if } \text{tl}: x \neq \phi
= <<x_1, y>, \ldots, <x_n, y>>
= \text{distr: } <x, y> \quad \square
\]