Reminder: Composition

\[ h(x) = f(g(x)) \]

or

\[ h(x_1, \ldots, x_n) = f(g_1(x_1, \ldots, x_n), \ldots, g_k(x_1, \ldots, x_n)). \]

Reminder: Recursion

\[ h(0) = k \]
\[ h(t + 1) = g(t, h(t)) \]

or

\[ h(x_1, \ldots, x_n, 0) = f(x_1, \ldots, x_n) \]
\[ h(x_1, \ldots, x_n, t + 1) = g(t, h(x_1, \ldots, x_n, t), x_1, \ldots, x_n). \]

Reminder: Initial Functions

\[ s(x) = x + 1 \]
\[ n(x) = 0 \]
\[ u_i^n(x_1, \ldots, x_n) = x_i, \ 1 \leq i \leq n. \]

The functions \( u_i^n \) are called projection functions. For example, \( u_2^3(x_1, x_2, x_3) = x_2 \).

Some Primitive Recursive Functions

**Example 1:** \( f(x, y) = x + y \)

We can transform this into a recursive definition:

\[ f(x, 0) = x \]
\[ f(x, y + 1) = f(x, y) + 1 \]

This can be rewritten as:

\[ f(x, 0) = u_1(x) \]
\[ f(x, y + 1) = g(y, f(x, y), x) \]

where \( g(x_1, x_2, x_3) = s(u_2^3(x_1, x_2, x_3)). \)

Obviously, \( u_1(x), u_2^3(x_1, x_2, x_3), \) and \( s(x) \) are primitive recursive functions – they are initial functions. \( g(x_1, x_2, x_3) \) is obtained by composition of primitive recursive functions, so it is primitive recursive itself. Therefore, \( f(x, y) = x + y \) is primitive recursive.

Some Primitive Recursive Functions

**Example 2:** \( h(x, y) = x \cdot y \)

We can obtain the following recursive definition:

\[ h(x, 0) = 0 \]
\[ h(x, y + 1) = h(x, y) + x \]

This can be rewritten as:

\[ h(x, 0) = n(x) \]
\[ h(x, y + 1) = g(y, h(x, y), x) \]

where

\[ f(x_1, x_2) = x_1 + x_2 \]
\[ g(x_1, x_2, x_3) = f(u_2^3(x_1, x_2, x_3), u_3^3(x_1, x_2, x_3)). \]

Obviously, these are all primitive recursive functions. Therefore, \( h(x, y) = x \cdot y \) is primitive recursive.

Some Primitive Recursive Functions

**Example 3:** \( h(x) = x! \)

Here are the recursion equations:

\[ h(0) = 1 \]
\[ h(t + 1) = t \cdot h(t) \]

This can be rewritten as:

\[ h(0) = s(n(t)) \]
\[ h(t + 1) = g(t, h(t)) \]

where

\[ g(x_1, x_2) = s(x_1) \cdot x_2, \] which can be written as:
\[ g(x_1, x_2) = s(u_1^2(x_1, x_2)) \cdot u_2^2(x_1, x_2). \]

Multiplication is already known to be primitive recursive. Therefore, \( h(x) = x! \) is primitive recursive.
Some Primitive Recursive Functions

In the following examples, we just show the recursive mechanism without developing the precise form of the recursion equations.

**Example 4:** \( x^y \)

The recursion equations are:

\[
\begin{align*}
x^0 & = 1 \\
x^{y+1} & = x^y \cdot x
\end{align*}
\]

**Example 5:** The predecessor function \( p(x) \)

It is defined as follows:

\[
\begin{align*}
p(x) & = x - 1 \quad \text{if } x \neq 0 \\
& = 0 \quad \text{if } x = 0
\end{align*}
\]

The recursion equations are:

\[
\begin{align*}
p(0) & = 0 \\
p(t + 1) & = t
\end{align*}
\]

**Example 6:** \( x - y \) (monus)

It is defined as follows:

\[
\begin{align*}
x - y & = x - y \quad \text{if } x \geq y \\
& = 0 \quad \text{if } x < y
\end{align*}
\]

The recursion equations are:

\[
\begin{align*}
x - 0 & = x \\
x - (t + 1) & = p(x - t)
\end{align*}
\]

**Example 7:** \(| x - y |\)

The function \(| x - y |\) is defined as the absolute value of the difference between \( x \) and \( y \).

It can be written as follows:

\[
|x - y| = (x - y) + (y - x)
\]

Therefore, \(| x - y |\) is primitive recursive.

**Example 8:** \( \alpha(x) \) ("negation")

The function \( \alpha(x) \) is defined as follows:

\[
\begin{align*}
\alpha(x) & = 1 \quad \text{if } x = 0 \\
& = 0 \quad \text{if } x \neq 0
\end{align*}
\]

\( \alpha(x) \) is primitive recursive, because:

\[
\alpha(x) = 1 - x
\]

If we prefer, we can just write down the recursion equations:

\[
\begin{align*}
\alpha(0) & = 1 \\
\alpha(t + 1) & = 0
\end{align*}
\]