mutual recursion

Canonical name	MutualRecursion
Date of creation	2013-03-22 19:06:31
Last modified on	2013-03-22 19:06:31
Owner	CWoo (3771)
Last modified by	CWoo (3771)
Numerical id	9
Author	CWoo (3771)
Entry type	Definition
Classification	msc 03D20
Synonym	simultaneous recursion

Mutual recursion is a way of defining functions via recursion involving several functions simultaneously. In mutual recursion, the value of the next argument of *any* function involved depends on values of the current arguments of *all* functions involved. The following are two simple examples:

- the pair of functions f, g such that $f(0) = 0$ and $g(0) = 1$, with $f(n+1) = f(n)g(n)$ and $g(n+1) = f(n) + g(n)$ is defined via mutual recursion. It is easy to see that $f(n) = 0$ and $g(n) = 1$.

- Fibonacci numbers can be interpreted via mutual recursion: $F(0) = 1$ and $G(0) = 1$, with $F(n+1) = F(n) + G(n)$ and $G(n+1) = F(n)$.

Formally,

Definition. Functions $f_1, \ldots, f_m : \mathbb{N}^{k+1} \to \mathbb{N}$ are said to be defined by *mutual recursion* via functions $g_1, \ldots g_m : \mathbb{N}^k \to \mathbb{N}$ and functions $h_1, \ldots, h_m : \mathbb{N}^{k+m+1} \to \mathbb{N}$, if

$$\begin{aligned} f_i(\boldsymbol{x}, 0) &= g_i(\boldsymbol{x}), \\ f_i(\boldsymbol{x}, n+1) &= h_i(\boldsymbol{x}, n, f_1(\boldsymbol{x}, n), \ldots, f_m(\boldsymbol{x}, n)), \end{aligned}$$

for any $\boldsymbol{x} \in \mathbb{N}^k$, and $i = 1, \ldots, m$.

Mutual recursion is an apparent generalization of primitive recursion, where the value of the next argument of a function depends on the value of the current argument of the same function, apparent because it is in fact equivalent to primitive recursion.

Proposition 1. *As above, if all of g_i and h_i are primitive recursive, so are all of f_i.*

Proof. We use the multiplicative encoding technique. Define function $F : \mathbb{N}^{k+1} \to \mathbb{N}$ as follows:

$$F(\boldsymbol{x}, n) = \exp(p_1, f_1(\boldsymbol{x}, n)) \cdots \exp(p_m, f_m(\boldsymbol{x}, n)),$$

where p_i is the i-th prime number ($p_1 = 2$, etc...). Furthermore,

$$(F(\boldsymbol{x}, n))_i = f_i(\boldsymbol{x}, n),$$

where $(z)_i$ denotes the exponent of prime p_i in z. Then

$$\begin{aligned}
F(\boldsymbol{x},0) &= \prod_{i=1}^{m} \exp(p_i, g_i(\boldsymbol{x})) = G(\boldsymbol{x}), \text{ and} \\
F(\boldsymbol{x},n+1) &= \prod_{i=1}^{m} \exp(p_1, f_1(\boldsymbol{x},n+1)) \\
&= \prod_{i=1}^{m} \exp(p_i, h_i(\boldsymbol{x},n,f_1(\boldsymbol{x},n),\ldots,f_m(\boldsymbol{x},n))) \\
&= \prod_{i=1}^{m} \exp(p_i, h_i(\boldsymbol{x},n,(F(\boldsymbol{x},n))_1,\ldots,(F(\boldsymbol{x},n))_m) \\
&= H(\boldsymbol{x},n,F(\boldsymbol{x},n))
\end{aligned}$$

where

$$G(\boldsymbol{x}) = \prod_{i=1}^{m} \exp(p_i, g_i(\boldsymbol{x})),$$

and

$$H(\boldsymbol{x},y,z) = \prod_{i=1}^{m} \exp(p_i, h_i(\boldsymbol{x},y,(z)_1,\ldots,(z)_m)$$

Since each of the g_i is primitive recursive, G is primitive recursive. Since each of the h_i is primitive recursive, H is primitive recursive. So F is primitive recursive, as it is defined via primitive recursion by G and H. Therefore, each $f_i = (F)_i$ is primitive recursive. □

Remark. The primitive recursiveness of mutual recursion can be derived from course-of-values recursion. The idea is the following: list the f_i's in order, and construct a single function F so the its values correspond to the values of the f_i's in the order given. For example, if two unary functions f_1 and f_2 defined by mutual recursion, then

$$\begin{array}{ccccccccccc}
f_1(0) & f_2(0) & f_1(1) & f_2(1) & f_1(2) & \cdots & f_1(k) & f_2(k) & f_1(k+1) & f_2(k+1) & \cdots \\
F(0) & F(1) & F(2) & F(3) & F(4) & \cdots & F(2k) & F(2k+1) & F(2k+2) & F(2k+3) & \cdots
\end{array}$$

Then it is not hard to see that $F(k+1)$ depends on $F(k), F(k-1)$, and $F(k-2)$, and an explicit formula can be derived expressing the dependency. Furthermore, this formula is easily seen to be primitive recursive, and hence so is $F(k)$.