1.3 Primitive Recursive Definitions

1.3.1 Primitive recursive definitions. Let $\rho[\vec{y},\vec{z}]$ and $\tau[\vec{y},x,a,\vec{z}]$ be terms containing at most the indicated variables free and neither of them applies the function symbol f. Then the functional equations

$$f(\vec{y},0,\vec{z}) = \rho[\vec{y},\vec{z}]$$
$$f(\vec{y},x+1,\vec{z}) = \tau[\vec{y},x,f(\vec{y},x,\vec{z}),\vec{z}]$$

has a unique solution f. The definition is called *primitive recursive definition* of f. The definition can be viewed as a function operator which takes all functions applied in the terms ρ and τ and yields the function f as a result. Note that we do not exclude the case when the parameters $\vec{y}$ or $\vec{z}$ or both are empty. Also the variable a does not have to occur freely in the term τ.

Example. Note that the operator of *iteration of unary function* is a special case of primitive recursive definitions. The operator takes a unary function f and yields a binary function $f^n(x)$ satisfying:

$$f^0(x) = x$$
$$f^{n+1}(x) = f\,f^n(x).$$

The function $f^n(x)$ is called the *iteration of* f. As a simple corollary of the next theorem we obtain that primitive recursive functions are closed also under iteration of unary functions.

1.3.2 Theorem *Primitive recursive functions are closed under primitive recursive definitions.*

Proof. Let f be defined by the primitive recursive definition as in Par. 1.3.1 from p.r. functions. First we define explicitly two auxiliary functions

$$g(w,\vec{y},\vec{z}) = \rho[\vec{y},\vec{z}]$$
$$h(x,a,w,\vec{y},\vec{z}) = \tau[\vec{y},x,a,\vec{z}],$$

which are primitive recursive by Thm. 1.2.3. Next we define a p.r. function f_1 by primitive recursion (note that we have at least one parameter!):

$$f_1(0,w,\vec{y},\vec{z}) = g(w,\vec{y},\vec{z})$$
$$f_1(\mathrm{S}(x),w,\vec{y},\vec{z}) = h\big(x,f_1(x,w,\vec{y},\vec{z}),w,\vec{y},\vec{z}\big).$$

We derive f as primitive recursive by the following explicit definition

$$f(\vec{y},x,\vec{z}) = f_1(x,0,\vec{y},\vec{z}).$$ □

1.3.3 Multiplication is primitive recursive. The multiplication function $x \times y$ is a p.r. function by the following primitive recursive definition:

$$0 \times y = 0$$
$$(x + 1) \times y = x \times y + y.$$

1.3.4 Exponentiation is primitive recursive. The exponentiation function x^y is a p.r. function by the following primitive recursive definition:

$$x^0 = 1$$
$$x^{y+1} = xx^y.$$

1.3.5 Summation function. The summation function $\sum_{i=0}^{n} i$ is a p.r. function by the following primitive recursive definition:

$$\sum_{i=0}^{0} i = 0$$
$$\sum_{i=0}^{n+1} i = \sum_{i=0}^{n} i + n + 1.$$

This is an example of *parameterless* primitive recursive definition.

1.3.6 Predecessor function is primitive recursive. The unary predecessor function $x \dot{-} 1$ is defined by the following *explicit definition with monadic discrimination on* x:

$$0 \dot{-} 1 = 0$$
$$x + 1 \dot{-} 1 = x.$$

The definition has a form of *parameterless* primitive recursive definition, where the term on the right hand side of the second identity is without any recursive application. Hence the predecessor function is primitive recursive.

1.3.7 Modified subtraction is primitive recursive. The modified subtraction function $x \dot{-} y$ is a p.r. function by primitive recursive definition:

$$x \dot{-} 0 = x$$
$$x \dot{-} (y + 1) = x \dot{-} y \dot{-} 1.$$

Note that the last occurrence of the symbol $\dot{-}$ in the second equation belongs to the application of the predecessor function.