Chapter II
Ordinary Recursion Theory

The notion of a recursive function resulted from an attempt in the 1930's to provide a precise mathematical characterization of the concept of a mechanically or algorithmically calculable function from ${}^k\omega$ into ω. One way to understand this concept is to imagine an idealized digital computer not subject to error or limitations of memory or storage space. Then a partial function F is mechanically calculable just in case there is a finite program (or algorithm) for this computer which directs it to accept inputs of the form $\mathbf{m}$ and carry out a computation with two possible results: if $\mathbf{m} \in \mathrm{Dm}\, F$, the computation terminates after finitely many steps with the correct value $F(\mathbf{m})$ as output; if $\mathbf{m} \notin \mathrm{Dm}\, F$, the computation does not terminate.

As this is an intuitive concept, however, it cannot be described completely except by convention. Not only is any attempt subject to legitimate disagreement on the basis of current knowledge, but also the possibility remains open that in the future a new means of calculation will be discovered which will be agreed by mathematicians to be mechanical but will not fall under the proposed description. Still, from a practical point of view, the notion seems to be a viable one: most people with a thorough understanding of the concepts involved will agree on the question of whether or not a given method of calculation is mechanical.

In particular, although we cannot give a rigorous proof that every recursive function is mechanically calculable, our justification of this assertion in § 2 below should be convincing to almost everyone. The converse proposition, known as Church's Thesis, that all mechanically calculable functions are recursive, is somewhat more problematic. Without a precise delineation of the class of mechanically calculable functions, we are in no position to *prove* that all of its members are recursive. We are forced, therefore, to rely on what might be called circumstantial evidence. Most importantly, no one has exhibited a function which is agreed to be mechanically calculable but is not recursive. In a similar vein, every known procedure which produces from calculable functions another calculable function also produces a recursive function from recursive functions.

Another kind of evidence is given by the variety of ways that the class of recursive functions can be characterized. Although these characterizations have quite different intuitive content (based on different conceptions of mechanical calculability) they all describe exactly the class of recursive functions. This shows that this is a very natural class and is at least intimately connected with the notion of mechanical calculability.

A discussion of these diverse characterizations and a more detailed examination of the evidence for Church's Thesis may be found in Kleene [1952].

As we shall be discussing functionals as well as functions, we shall want, for comparison, also a notion of mechanical calculability for partial functions from ${}^{k,l}\omega$ into ω. At first glance there seems to be no way for our idealized computer to accept inputs of the form $(\mathbf{m}, \boldsymbol{\alpha})$. Even if we allow the computer to have infinite memory facilities sufficient to store all the values of an argument α, it would seemingly take infinitely long just to "read in" these values. Hence to preserve the finiteness of computations we say that the computer receives an input $(\mathbf{m}, \boldsymbol{\alpha})$ when it is connected to an infinite memory device in which have previously been stored $m_0, \ldots, m_{k-1}$ and the complete graphs of $\alpha_0, \ldots, \alpha_{l-1}$. The computer may then refer to this device at any point in the computation to transfer to its working "registers" either an m_i or a value $\alpha_j(p)$. Since the computation of a value $\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha})$ must be finite, only finitely many values of each argument are actually used. Thus mechanically calculable functionals are continuous.

1. Primitive Recursion

We examine first the class of primitive recursive functionals. We shall show that this class includes many familiar functionals but fails to exhaust the class of mechanically calculable functionals. Although in this section we are concerned only with total functionals, we state some of the definitions with '$\simeq$' rather than '$=$' for future application to partial functionals.

1.1 Definition. For any k, l, and n, any $i < k$ and $j < l$, and any $(\mathbf{m}, \boldsymbol{\alpha}) \in {}^{k,l}\omega$,

(i) (*the initial functionals*)

$$\mathsf{Cs}_n^{k,l}(\mathbf{m}, \boldsymbol{\alpha}) = n, \qquad \mathsf{Pr}_i^{k,l}(\mathbf{m}, \boldsymbol{\alpha}) = m_i,$$

$$\mathsf{Sc}_i^{k,l}(\mathbf{m}, \boldsymbol{\alpha}) = m_i + 1, \quad \text{and} \quad \mathsf{Ap}_{i,j}^{k,l}(\mathbf{m}, \boldsymbol{\alpha}) = \alpha_j(m_i);$$

(ii) (*functional composition*) for any k' and any functionals $\mathsf{G}, \mathsf{H}_0, \ldots, \mathsf{H}_{k'-1}$, $\mathsf{FCmp}_{k'}^{k,l}(\mathsf{G}, \mathsf{H}_0, \ldots, \mathsf{H}_{k'-1})$ is the functional F of rank (k, l) such that

(a) if G is of rank (k', l) and $\mathsf{H}_0, \ldots, \mathsf{H}_{k'-1}$ are all of rank (k, l), then

$$\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}) \simeq \mathsf{G}(\mathsf{H}_0(\mathbf{m}, \boldsymbol{\alpha}), \ldots, \mathsf{H}_{k'-1}(\mathbf{m}, \boldsymbol{\alpha}), \boldsymbol{\alpha});$$

(b) otherwise, $\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}) \simeq 0$;

(iii) (*primitive recursion*) for any functionals G and H, $\mathrm{Rec}^{k+1,l}(\mathsf{G},\mathsf{H})$ is the functional F of rank $(k+1, l)$ such that

(a) if G is of rank (k, l) and H is of rank $(k+2, l)$, then $\mathsf{F}(0, \mathbf{m}, \boldsymbol{\alpha}) \simeq \mathsf{G}(\mathbf{m}, \boldsymbol{\alpha})$, and for all p,

$$\mathsf{F}(p+1, \mathbf{m}, \boldsymbol{\alpha}) \simeq \mathsf{H}(\mathsf{F}(p, \mathbf{m}, \boldsymbol{\alpha}), p, \mathbf{m}, \boldsymbol{\alpha});$$

(b) otherwise, $\mathsf{F}(p, \mathbf{m}, \boldsymbol{\alpha}) \simeq 0$.

1.2 Definition. The class Prf of *primitive recursive functionals* is the smallest class of total functionals which contains the initial functionals and is closed under functional composition and primitive recursion.

Note that Prf is inductively defined by closure under finitary functions. By induction over Prf it follows that every primitive recursive functional is total and mechanically calculable: clearly this is true for the initial functionals and these properties are preserved by functional composition and primitive recursion.

1.3 Examples. The *addition* function $(+)$ is defined by the equations $0 + m = n$ and $(p+1) + m = (p+m) + 1$. A simple calculation shows that

$$+ = \mathrm{Rec}^{2,0}(\mathsf{Pr}_0^{1,0}, \mathsf{Sc}_0^{3,0})$$

and is thus primitive recursive. *Multiplication* satisfies $0 \cdot m = 0$ and $(p+1) \cdot m = p \cdot m + m$, so that

$$\cdot = \mathrm{Rec}^{2,0}(\mathsf{Cs}_0^{1,0}, \mathrm{FCmp}_2^{3,0}(+, \mathsf{Pr}_0^{3,0}, \mathsf{Pr}_2^{3,0}))$$

and is thus primitive recursive. The *exponential* function $\exp(p, m) = m^p$ satisfies $\exp(0, m) = 1$ and $\exp(p+1, m) = \exp(p, m) \cdot m$ and is similarly shown to be primitive recursive. The *factorial* function (!) satisfies $0! = 1$ and $(p+1)! = p!(p+1)$ and is primitive recursive. Let

$$\mathrm{sg}^+(p) = \begin{cases} 0, & \text{if } p = 0; \\ 1, & \text{if } p > 0; \end{cases} \quad \text{and} \quad \mathrm{sg}^-(p) = \begin{cases} 1, & \text{if } p = 0; \\ 0, & \text{if } p > 0. \end{cases}$$

Then

$$\mathrm{sg}^+ = \mathrm{Rec}^{1,0}(\mathsf{Cs}_0^{0,0}, \mathsf{Cs}_1^{2,0}) \quad \text{and} \quad \mathrm{sg}^- = \mathrm{Rec}^{1,0}(\mathsf{Cs}_1^{0,0}, \mathsf{Cs}_0^{2,0}),$$

so both are primitive recursive. Let f be the primitive recursive function $\mathrm{Rec}^{1,0}(\mathsf{Cs}_0^{0,0}, \mathsf{Pr}_1^{2,0})$ so that $f(0) = 0$ and $f(p+1) = p$ (the *predecessor* function). Then if we set

$$g = \mathrm{Rec}^{2,0}(\mathsf{Pr}_1^{1,0}, \mathrm{FCmp}^{3,0}(f, \mathsf{Pr}_0^{3,0})),$$

it is straightforward to check that

$$g(p, m) = \begin{cases} m - p, & \text{if} \quad m \geqslant p; \\ 0, & \text{otherwise.} \end{cases}$$

$g(p, m)$ is usually written $m \dot{-} p$.

We call a relation R *primitive recursive* just in case its characteristic functional K_R is primitive recursive. Then $\mathsf{K}_\leqslant(m, p) = \mathrm{sg}^+(m \dot{-} p)$, $\mathsf{K}_\geqslant(m, p) = \mathrm{sg}^+(p \dot{-} m)$, $\mathsf{K}_<(m, p) = \mathrm{sg}^-(p \dot{-} m)$, $\mathsf{K}_>(m, p) = \mathrm{sg}^-(m \dot{-} p)$, and $\mathsf{K}_=(m, p) = \mathrm{sg}^+(\mathsf{K}_\leqslant(m, p) + \mathsf{K}_\geqslant(m, p))$ so these relations are all primitive recursive. Furthermore, if R and S are primitive recursive relations of the same rank, then

$$\mathsf{K}_{\mathsf{R}\cup\mathsf{S}}(\mathbf{m}, \boldsymbol{\alpha}) = \mathsf{K}_\mathsf{R}(\mathbf{m}, \boldsymbol{\alpha}) \cdot \mathsf{K}_\mathsf{S}(\mathbf{m}, \boldsymbol{\alpha}),$$

$$\mathsf{K}_{\mathsf{R}\cap\mathsf{S}}(\mathbf{m}, \boldsymbol{\alpha}) = \mathrm{sg}^+(\mathsf{K}_\mathsf{R}(\mathbf{m}, \boldsymbol{\alpha}) + \mathsf{K}_\mathsf{S}(\mathbf{m}, \boldsymbol{\alpha})), \quad \text{and}$$

$$\mathsf{K}_{\sim\mathsf{R}}(\mathbf{m}, \boldsymbol{\alpha}) = 1 \dot{-} \mathsf{K}_\mathsf{R}(\mathbf{m}, \boldsymbol{\alpha})$$

so that the class of primitive recursive relations of a given rank forms a Boolean algebra.

Our next aim is to show that the sequence coding and decoding functions of § I.1.4 are primitive recursive. To this end we establish some further closure properties of the classes of primitive recursive functionals and relations.

1.4 Definition. For any k, l, k', and l', any functionals $\mathsf{G}, \mathsf{G}_0, \ldots, \mathsf{G}_{k'}$, and any relations $\mathsf{R}_0, \ldots, \mathsf{R}_{k'-1}$, and S,

(i) (*expansion*) if G has rank (k, l), then $\mathrm{Ex}_{k',l'}(\mathsf{G})$ is the functional F of rank $(k + k', l + l')$ such that

$$\mathsf{F}(\mathbf{m}, \mathbf{n}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \simeq \mathsf{G}(\mathbf{m}, \boldsymbol{\alpha});$$

(ii) (*bounded search*) if G has rank $(k + 2, l)$, then $\mathrm{Bs}(\mathsf{G})$ is the functional F of rank $(k + 1, l)$ such that

$$\mathsf{F}(p, \mathbf{m}, \boldsymbol{\alpha}) \simeq \begin{cases} q, & \text{if} \quad q < p,\ \mathsf{G}(q, \mathbf{m}, \boldsymbol{\alpha}) \simeq 0, \quad \text{and} \\ & \qquad (\forall r < q)(\exists n > 0) \,.\, \mathsf{G}(r, \mathbf{m}, \boldsymbol{\alpha}) \simeq n; \\ p, & \text{if} \quad (\forall q < p)(\exists n > 0) \,.\, \mathsf{G}(q, \mathbf{m}, \boldsymbol{\alpha}) \simeq n; \end{cases}$$

we write

$$\mathsf{F}(p, \mathbf{m}, \boldsymbol{\alpha}) \simeq \text{"least"} \quad q < p \,.\, \mathsf{G}(q, \mathbf{m}, \boldsymbol{\alpha}) \simeq 0;$$

(iii) (*definition by cases*) if $\mathsf{G}_0, \dots, \mathsf{G}_{k'}$, $\mathsf{R}_0, \dots, \mathsf{R}_{k'-1}$ all have rank (k, l) and for any $(\mathbf{m}, \boldsymbol{\alpha}) \in {}^{k,l}\omega$ there is at most one $i < k'$ such that $\mathsf{R}_i(\mathbf{m}, \boldsymbol{\alpha})$, then $\mathrm{Cases}_{k'}(\mathsf{G}_0, \dots, \mathsf{G}_{k'}, \mathsf{R}_0, \dots, \mathsf{R}_{k'-1})$ is the functional F of rank (k, l) such that

$$\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}) \simeq \begin{cases} \mathsf{G}_0(\mathbf{m}, \boldsymbol{\alpha}), & \text{if } \mathsf{R}_0(\mathbf{m}, \boldsymbol{\alpha}); \\ \quad\vdots & \\ \mathsf{G}_{k'-1}(\mathbf{m}, \boldsymbol{\alpha}), & \text{if } \mathsf{R}_{k'-1}(\mathbf{m}, \boldsymbol{\alpha}); \\ \mathsf{G}_{k'}(\mathbf{m}, \boldsymbol{\alpha}), & \text{otherwise } \ (\forall i < k' \sim \mathsf{R}_i(\mathbf{m}, \boldsymbol{\alpha})); \end{cases}$$

(iv) (*relational composition*) if $\mathsf{G}_0, \dots, \mathsf{G}_{k'-1}$ have rank (k, l) and S has rank (k', l), then $\mathrm{RCmp}_{k'}(\mathsf{S}, \mathsf{G}_0, \dots, \mathsf{G}_{k'-1})$ is the relation R of rank (k, l) such that

$$\mathsf{R}(\mathbf{m}, \boldsymbol{\alpha}) \leftrightarrow \mathsf{S}(\mathsf{G}_0(\mathbf{m}, \boldsymbol{\alpha}), \dots, \mathsf{G}_{k'-1}(\mathbf{m}, \boldsymbol{\alpha}), \boldsymbol{\alpha});$$

(v) (*bounded quantification*) if S is of rank $(k+2, l)$, then $\exists^0_<(\mathsf{S})$ and $\forall^0_<(\mathsf{S})$ are the relations P and Q of rank $(k+1, l)$ such that

$$\mathsf{P}(p, \mathbf{m}, \boldsymbol{\alpha}) \leftrightarrow (\exists q < p)\mathsf{S}(q, \mathbf{m}, \boldsymbol{\alpha}), \quad \text{and}$$

$$\mathsf{Q}(p, \mathbf{m}, \boldsymbol{\alpha}) \leftrightarrow (\forall q < p)\mathsf{S}(q, \mathbf{m}, \boldsymbol{\alpha}).$$

1.5 Theorem. *The class of primitive recursive functionals and relations is closed under expansion, bounded search, definition by cases, relational composition, and bounded quantification.*

Proof. (i) Clearly any expansion of an initial functional is still an initial functional. Any expansion of $\mathrm{FCmp}(\mathsf{G}, \mathsf{H}_0, \dots, \mathsf{H}_{k'-1})$ is $\mathrm{FCmp}(\mathsf{G}', \mathsf{H}'_0, \dots, \mathsf{H}'_{k'-1})$ for suitable expansions G' and H'_i of G and H_i. Similarly, any expansion of $\mathrm{Rec}(\mathsf{G}, \mathsf{H})$ is $\mathrm{Rec}(\mathsf{G}', \mathsf{H}')$ for suitable G' and H'. Hence by induction the expansion of any primitive recursive functional is primitive recursive.

(ii) If G is primitive recursive, then so is F defined by:

$$\mathsf{F}(0, \mathbf{m}, \boldsymbol{\alpha}) = 0;$$

$$\mathsf{F}(p+1, \mathbf{m}, \boldsymbol{\alpha}) = \mathsf{F}(p, \mathbf{m}, \boldsymbol{\alpha}) + \mathrm{sg}^+(\mathsf{G}(\mathsf{F}(p, \mathbf{m}, \boldsymbol{\alpha}), \mathbf{m}, \boldsymbol{\alpha})).$$

We leave to the reader the amusing verification that F is $\mathrm{Bs}(\mathsf{G})$.

(iii) Suppose that $\mathsf{G}_0, \dots, \mathsf{G}_{k'}$ and $\mathsf{R}_0, \dots, \mathsf{R}_{k'-1}$ satisfy the hypothesis for definition by cases. Then the F defined there is also given by

$$\begin{aligned} \mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}) = {} & [\mathsf{G}_0(\mathbf{m}, \boldsymbol{\alpha}) \cdot \mathrm{sg}^-(\mathsf{K}_{\mathsf{R}_0}(\mathbf{m}, \boldsymbol{\alpha}))] + \cdots + [\mathsf{G}_{k'-1}(\mathbf{m}, \boldsymbol{\alpha}) \cdot \mathrm{sg}^-(\mathsf{K}_{\mathsf{R}_{k'-1}}(\mathbf{m}, \boldsymbol{\alpha}))] \\ & + [\mathsf{G}_{k'}(\mathbf{m}, \boldsymbol{\alpha}) \cdot \mathsf{K}_{\mathsf{R}_0}(\mathbf{m}, \boldsymbol{\alpha}) \cdot \cdots \cdot \mathsf{K}_{\mathsf{R}_{k'-1}}(\mathbf{m}, \boldsymbol{\alpha})], \end{aligned}$$

and is thus seen to be primitive recursive.

(iv) If $\mathsf{R} = \mathrm{RCmp}(\mathsf{S}, \mathsf{G}_0, \ldots, \mathsf{G}_{k'-1})$, then $\mathsf{K}_\mathsf{R} = \mathrm{FCmp}(\mathsf{K}_\mathsf{S}, \mathsf{G}_0, \ldots, \mathsf{G}_{k'-1})$; so if $\mathsf{S}, \mathsf{G}_0, \ldots,$ and $\mathsf{G}_{k'-1}$ are all primitive recursive, so is R.

(v) Let P, Q, and S be as in the definition. Then

$$\mathsf{P}(p, \mathbf{m}, \boldsymbol{\alpha}) \leftrightarrow [\text{“least”}\ q < p\,.\,\mathsf{S}(q, \mathbf{m}, \boldsymbol{\alpha})] < p,$$

and $\mathsf{Q} = {\sim}\exists^0_<({\sim}\mathsf{S})$. $\square$

These results will be used, usually without reference, to justify the claim that some explicitly defined functional or relation is primitive recursive. For example, if G, H, I, R, S, and T are primitive recursive, and F is defined by:

$$\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}) = \begin{cases} m_0 + \alpha_2(m_1), & \text{if} \quad (\exists q < s)\mathsf{R}(q, \mathsf{G}(q, \mathbf{m}, \boldsymbol{\alpha}), \alpha_4); \\ \alpha_1(\text{“least”}\ q < \mathsf{H}(m_3, \alpha_0, \alpha_2) \\ \qquad\qquad [q \cdot m_1 \geqslant \mathsf{I}(m_2, \alpha_3)]), & \text{if} \quad \mathsf{S}(\mathbf{m}, \boldsymbol{\alpha}) \wedge \neg\mathsf{T}(m_0, \boldsymbol{\alpha}); \\ 0, \quad \text{otherwise}; \end{cases}$$

then a succession of applications of the clauses of Theorem 1.5 together with the remarks preceding it shows that F is primitive recursive.

As a first application, we obtain the primitve recursiveness of the relation “m divides p” and the function $\langle p_m : m \in \omega \rangle$ which enumerates the prime numbers:

$$m \text{ divides } p \leftrightarrow (\exists q < p + 1)(q \cdot m = p);$$

$$p_0 = 2, \quad \text{and}$$

$$p_{m+1} = (\text{“least”}\ q < p_m! + 2)[p_m < q \wedge (\neg\exists r < q)(1 < r \wedge r \text{ divides } q)].$$

1.6 Corollary. *The sequence coding and decoding functions and the set* Sq *of* §I.1.4 *are all primitive recursive.*

Proof. For the functions, this is immediate from their definitions. Also $s \in \mathrm{Sq} \leftrightarrow (\forall i < s)[p_i \text{ divides } s \rightarrow i < \lg(s)]$. $\square$

For any functional F of rank $(k + 1, l)$ we set

$$\bar{\mathsf{F}}(0, \mathbf{m}, \boldsymbol{\alpha}) = \langle\ \rangle \quad \text{and}$$

$$\bar{\mathsf{F}}(p + 1, \mathbf{m}, \boldsymbol{\alpha}) = \bar{\mathsf{F}}(p, \mathbf{m}, \boldsymbol{\alpha}) * \langle \mathsf{F}(p, \mathbf{m}, \boldsymbol{\alpha})\rangle.$$

Thus $\bar{\mathsf{F}}(p, \mathbf{m}, \boldsymbol{\alpha}) \simeq \langle \mathsf{F}(0, \mathbf{m}, \boldsymbol{\alpha}), \ldots, \mathsf{F}(p-1, \mathbf{m}, \boldsymbol{\alpha})\rangle$. From the definition and the preceding Corollary it is clear that if F is primitive recursive, so is $\bar{\mathsf{F}}$. Furthermore, for any $q > p$, $\mathsf{F}(p, \mathbf{m}, \boldsymbol{\alpha}) \simeq (\bar{\mathsf{F}}(q, \mathbf{m}, \boldsymbol{\alpha}))_p$ so the primitive recursiveness of $\bar{\mathsf{F}}$ implies that of F. The same argument shows that the functional $\mathsf{H}(p, \alpha) = \bar{\alpha}(p)$ is primitive recursive.

1.7 Definition (*course-of-values recursion*). For any k and l, and any functional G of rank $(k+2, l)$, CvRec(G) is the functional F of rank $(k+1, l)$ such that

$$\mathsf{F}(p, \mathbf{m}, \boldsymbol{\alpha}) \simeq \mathsf{G}(\bar{\mathsf{F}}(p, \mathbf{m}, \boldsymbol{\alpha}), p, \mathbf{m}, \boldsymbol{\alpha}).$$

1.8 Theorem. *The class of primitive recursive functionals is closed under course-of-values recursion.*

Proof. If G is of rank $(k+2, l)$ and $\mathsf{F} = \mathrm{CvRec}(\mathsf{G})$, then

$$\bar{\mathsf{F}}(p+1, \mathbf{m}, \boldsymbol{\alpha}) = \bar{\mathsf{F}}(p, \mathbf{m}, \boldsymbol{\alpha}) * \langle \mathsf{G}(\bar{\mathsf{F}}(p, \mathbf{m}, \boldsymbol{\alpha}), p, \mathbf{m}, \boldsymbol{\alpha})\rangle.$$

Hence, if G is primitive recursive, so is $\bar{\mathsf{F}}$. But then by the preceding remarks, also F is primitive recursive. □

In applying these theorems to show that a particular relation R is primitive recursive, we must formally work with K_R and show this to be a primitive recursive functional. Usually, however, it is more perspicuous to describe directly recursive conditions on R. For example, the condition

$$R(p) \leftrightarrow p = 0 \vee (\exists q < p)[q + 7 = p \wedge R(q)]$$

is equivalent to

$$\mathsf{K}_R(p) = \begin{cases} 0, & \text{if } \ p = 0; \\ F(p, \bar{\mathsf{K}}_R(p)), & \text{if } \ p > 0; \end{cases}$$

where

$$F(p, s) = \begin{cases} 0, & \text{if } \ (\exists q < p)[q + 7 = p \wedge (s)_q = 0]; \\ 1, & \text{otherwise.} \end{cases}$$

In such cases we shall leave to the reader the translation of the conditions on R to conditions on K_R.

Another technique we shall use frequently is to give definitions of the form

$$\mathsf{F}(i, j, \langle \mathbf{m} \rangle, \langle \boldsymbol{\alpha} \rangle) = \mathsf{G}(\alpha_j(m_i), \langle \mathbf{m} \rangle, \langle \boldsymbol{\alpha} \rangle).$$

This should be taken as an abbreviation for

$$\mathsf{F}(i, j, s, \gamma) = \mathsf{G}((\gamma((s)_i))_j, s, \gamma)$$

so that F is defined for all arguments, not only those of the form $\langle \mathbf{m} \rangle$ and $\langle \boldsymbol{\alpha} \rangle$. Since the decoding functions are primitive recursive, if G is primitive recursive, so is F.

We turn now to the assertion that not all mechanically calculable functionals are primitive recursive. Our method derives from the description of mechanical computability in terms of an idealized computer. We shall in effect specify a particular computer and a "programming language" which suffices to write programs for computing all primitive recursive functionals. We can then exhibit a functional which is mechanically calculable but cannot be "programmed" in this language and hence is not primitive recursive.

The "language" is simply a set $\mathrm{Pri} \subseteq \omega$, members of which we call *primitive recursive indices.* To each $a \in \mathrm{Pri}$ is assigned a primitive recursive functional $[a]$ by interpreting a as a program for an idealized computer whose basic operations correspond to the clauses of Definition 1.1.

1.9 Definition. Pri is the smallest subset of ω such that for all k, l, and n, all $i < k$, and all $j < l$,
(0) $\langle 0, k, l, 0, n \rangle$, $\langle 0, k, l, 1, i \rangle$, $\langle 0, k, l, 2, i \rangle$, and $\langle 0, k, l, 3, i, j \rangle$ all belong to Pri;
(1) for any k' and any $b, c_0, \ldots, c_{k'-1} \in \mathrm{Pri}$, $\langle 1, k, l, b, c_0, \ldots, c_{k'-1} \rangle \in \mathrm{Pri}$;
(2) for any $b, c \in \mathrm{Pri}$, $\langle 2, k+1, l, b, c \rangle \in \mathrm{Pri}$.

This is clearly a monomorphic inductive definition, so by Theorem I.3.5 there exists a unique map $[\cdot]$ from Pri into the class of functionals such that

(0) $[\langle 0, k, l, 0, n \rangle] = \mathsf{Cs}_n^{k,l}$;

$[\langle 0, k, l, 1, i \rangle] = \mathsf{Pr}_i^{k,l}$;

$[\langle 0, k, l, 2, i \rangle] = \mathsf{Sc}_i^{k,l}$;

$[\langle 0, k, l, 3, i, j \rangle] = \mathsf{Ap}_i^{k,l}$;

(1) $[\langle 1, k, l, b, c_0, \ldots, c_{k'-1} \rangle] = \mathrm{FCmp}_{k'}^{k,l}([b], [c_0], \ldots, [c_{k'-1}])$;
(2) $[\langle 2, k+1, l, b, c \rangle] = \mathrm{Rec}^{k,l}([b], [c])$.

1.10 Theorem. $\mathrm{Prf} = \{[a]\colon a \in \mathrm{Pri}\}$.

Proof. For the inclusion ($\subseteq$) we observe that $\{[a]\colon a \in \mathrm{Pri}\}$ clearly contains the initial functions and is closed under composition and primitive recursion. For ($\supseteq$) consider $\{a\colon a \in \mathrm{Pri} \wedge [a]$ is primitive recursive$\}$. This set satisfies clauses (0)–(2) of Definition 1.9 and thus includes Pri. $\square$

For each k and l, set

$$\mathsf{Ev}^{k,l}(a, \mathbf{m}, \boldsymbol{\alpha}) = \begin{cases} [a](\mathbf{m}, \boldsymbol{\alpha}), & \text{if} \quad a \in \mathrm{Pri} \wedge (a)_1 = k \wedge (a)_2 = l; \\ 0, & \text{otherwise.} \end{cases}$$

$\mathsf{Ev}^{k,l}$ is called an *evaluation function.*

1.11 Theorem. *for all* $k > 0$ *and all* l, $\mathsf{Ev}^{k,l}$ *is mechanically calculable but not primitive recursive.*

Proof. Suppose first that $\mathsf{Ev}^{k+1,l}$ were primitive recursive. Then if

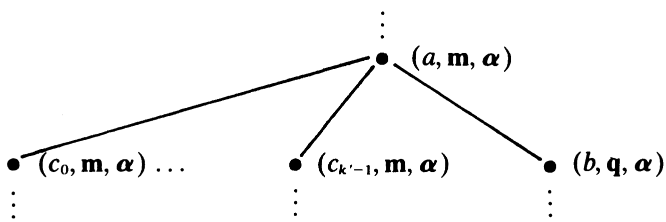
$$\mathsf{F}(a, \mathbf{m}, \boldsymbol{\alpha}) = \mathsf{Ev}^{k+1,l}(a, a, \mathbf{m}, \boldsymbol{\alpha}) + 1$$

also F is primitive recursive. Hence by Theorem 1.10, $\mathsf{F} = [b]$ for some $b \in \mathrm{Pri}$. But then

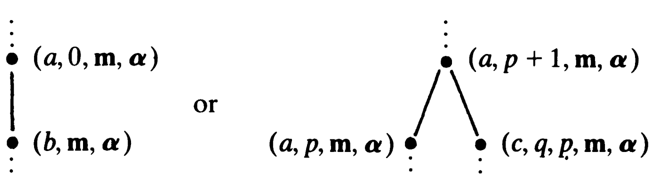
$$\mathsf{F}(b, \mathbf{m}, \boldsymbol{\alpha}) = \mathsf{Ev}^{k+1,l}(b, b, \mathbf{m}, \boldsymbol{\alpha}) + 1 = [b](b, \mathbf{m}, \boldsymbol{\alpha}) + 1 = \mathsf{F}(b, \mathbf{m}, \boldsymbol{\alpha}) + 1,$$

a contradiction.

To see that $\mathsf{Ev}^{k,l}$ is mechanically calculable we examine the notion of a *computation tree.* Such a tree may be thought of as a schematic representation of the action of the idealized computer in calculating a value of a given functional. Each computation tree has a top node x_0. Each node x has 0 or more *immediate predecessors* which lie just below x. Each node x is labeled with a triple $(a, \mathbf{m}, \boldsymbol{\alpha})$. x is said to be *evaluated* when $[a](\mathbf{m}, \boldsymbol{\alpha})$ is computed. If a is an index for one of the initial functions, then x has no immediate predecessors. If $a = \langle 1, k, l, b, c_0, \ldots, c_{k'-1} \rangle$, then x has $k' + 1$ immediate predecessors labeled as follows:



where $\mathbf{q} = (q_0, \ldots, q_{k'-1})$ and for all $i < k'$, $[c_i](\mathbf{m}, \boldsymbol{\alpha}) = q_i$. If $a = \langle 2, k+1, l, b, c \rangle$, then x has either 1 or 2 immediate predecessors:



where $[a](p, \mathbf{m}, \boldsymbol{\alpha}) = q$. Thus the labels on the immediate predecessors of x correspond to the *subcomputations* necessary to evaluate the label at x.

For any triple $(a, \mathbf{m}, \boldsymbol{\alpha})$ with $a \in \text{Pri}$, $(a)_1 = k$, and $(a)_2 = l$, we generate and label a computation tree as follows. The top node is labeled $(a, \mathbf{m}, \boldsymbol{\alpha})$. Depending on a, the appropriate number of immediate predecessors of x_0 are constructed and all but possibly the right-most one labeled. The number of immediate predecessors of these may then be determined and so on. If node x lies below node y, then the index at x is not greater than the index at y, and if they are equal, then the first argument at x is strictly less than the first argument at y. Hence each branch terminates with a node labeled with an index for one of the initial functions. This is immediately evaluable. If at some stage all nodes at a given level except the right-most one have been evaluated, then this one may be labeled. When all immediate predecessors of a given node x have been evaluated, then x may be evaluated and has the value of the right-most immediate predecessor as its value. In any application of composition, $k' < a$. Hence by the Infinity Lemma (I.2.3) the tree is finite and this process terminates after a finite number of steps with an evaluation of the top node x_0 and hence with the value $\mathsf{Ev}^{k,l}(a, \mathbf{m}, \boldsymbol{\alpha})$.

A mechanical procedure for calculating $\mathsf{Ev}^{k,l}(a, \mathbf{m}, \boldsymbol{\alpha})$ now goes as follows. Determine first whether or not $a \in \text{Pri}$, $(a)_1 = k$, and $(a)_2 = l$. This is possible by Corollary 1.6 and Exercise 1.16. If not, the value is 0. If so, construct the computation tree as described above and read off the value of the top node. □

1.12–1.18 Exercises

1.12. Find explicitly primitive recursive indices for addition and multiplication.

1.13. Show that any primitive recursive function has infinitely many indices.

1.14. Show that if G is primitive recursive and

$$\mathsf{F}(p, \mathbf{m}, \boldsymbol{\alpha}) = \mathsf{G}(p, \mathbf{m}, \boldsymbol{\alpha}, \lambda q \,.\, \mathsf{F}_p(q, \mathbf{m}, \boldsymbol{\alpha}))$$

where

$$\mathsf{F}_p(q, \mathbf{m}, \boldsymbol{\alpha}) = \begin{cases} \mathsf{F}(q, \mathbf{m}, \boldsymbol{\alpha}), & \text{if } q < p; \\ 0, & \text{otherwise;} \end{cases}$$

then also F is primitive recursive.

1.15. Show that if G and H are both primitive recursive and

$$\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}) = \mathsf{G}(\mathbf{m}, \boldsymbol{\alpha}, \lambda p \,.\, \mathsf{H}(p, \mathbf{m}, \boldsymbol{\alpha})),$$

then F is primitive recursive.

1.16. Show that the set Pri of primitive recursive indices is a primitive recursive set.

1.17. For each of the closure properties established above, there is a stronger *effective closure property* which asserts that an index for the resulting functional or relation can be computed from indices of the component functionals and relations. For example, Prf is *effectively closed* under bounded search iff there exists a primitive recursive function f such that for all $a \in \text{Pri}$, $[f(a)] = \text{Bs}([a])$. Show that the class of primitive recursive functionals and relations is effectively closed under expansion, bounded search, and course-of-values recursion.

1.18. For any $\boldsymbol{\beta} = (\beta_0, \ldots, \beta_{r-1})$, let Prf[$\boldsymbol{\beta}$], the class of functionals *primitive recursive in* $\boldsymbol{\beta}$, be the smallest class of total functionals which contains $\beta_0, \ldots, \beta_{r-1}$, and the initial functionals, and is closed under functional composition and primitive recursion. Show that $\mathsf{F} \in \text{Prf}[\boldsymbol{\beta}]$ iff, for some primitive recursive G, $\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}) = \mathsf{G}(\mathbf{m}, \boldsymbol{\alpha}, \boldsymbol{\beta})$.

1.19 Notes. The primitive recursive functions have played a varying role over the years. Gödel [1931] used "rekursiv" to mean primitive recursive and used only these functions in the incompleteness proof. As various formalized analogues of the class of mechanically calculable functions came to be studied in the 1930's and the class of (general) recursive functions was introduced, the primitive recursive functions were relegated to the secondary role of providing a basic stock of methods for elementary calculations (cf. Exercise 1.17 and Corollary 3.3 below). That some kinds of "recursions" lead beyond the primitive recursive functions was already shown in Ackermann [1928] and by the method of Theorem 1.11 in Péter [1935]. Péter [1967] has a good deal of information on various extensions of the class of primitive recursive functions by means of more complicated recursions.

The method of proof used in Theorem 1.11 is called a *diagonal argument* and is used repeatedly in the remainder of the book. It is an adaptation of Cantor's proof that the set of real numbers is uncountable.

2. *Recursive Functionals and Relations*

If the class of recursive functionals is to include all mechanically calculable functionals, then it must evidently contain each $\mathsf{Ev}^{k,l}$. This suggests that we consider the class Prf_1, the smallest class of total functionals which contains the initial functionals and all $\mathsf{Ev}^{k,l}$ and is closed under functional composition and primitive recursion. Unfortunately, the methods of §1 are easily adapted to show that there is an indexing $[\cdot]_1$ of Prf_1 such that the functionals

$$\mathsf{Ev}_1^{k,l}(a, \mathbf{m}, \boldsymbol{\alpha}) = \begin{cases} [a]_1(\mathbf{m}, \boldsymbol{\alpha}), & \text{if} \quad a \in \mathrm{Pri}_1, \quad (a)_1 = k, \quad \text{and} \quad (a)_2 = l; \\ 0, & \text{otherwise;} \end{cases}$$

are mechanically calculable but not in Prf_1.

It is still instructive to examine the properties of Prf_1 more closely. For any $\mathsf{G} \in \mathrm{Prf}_1$, the functional F defined by

$$\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}) = \mathsf{Ev}^{k,l}(\mathsf{G}(\mathbf{m}, \boldsymbol{\alpha}), \mathbf{m}, \boldsymbol{\alpha})$$

is also in Prf_1. The computation of $\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha})$ proceeds in two parts: first a value $\mathsf{G}(\mathbf{m}, \boldsymbol{\alpha})$ is determined, then this value is interpreted as a "program" for the rest of the computation. The principle that the results of one part of a computation may be interpreted as coded instructions for another part we call the *self-determination principle.* It is an idealized form of the various devices for program modification which play an essential role in most programming languages for electronic computers.

The class Prf_1 exhibits the self-determination principle only in a limited form. Although the functional G may be any element of Prf_1, the remainder of the computation described above is determined by an index in Pri (not Pri_1) and thus cannot make further use of the functionals $\mathsf{Ev}^{k,l}$. We should like to allow for the possibility that the result $\mathsf{G}(\mathbf{m}, \boldsymbol{\alpha})$ of one computation might be interpreted as an index for any functional in the class under consideration (*full self-determination principle*).

To accomplish this we change slightly our point of view. Instead of fixing in advance a class of indices which code instructions for computing total functionals, we shall regard every natural number a as an index for a *partial* functional $\{a\}$. As before we shall have $\{a\}(\mathbf{m}, \boldsymbol{\alpha}) \simeq n$ just in case a codes a "program" which, when applied to the arguments $(\mathbf{m}, \boldsymbol{\alpha})$ leads to the value n. Now, however, a program may lead to a value for some arguments but not for others. Indeed many a (0, for example) are not of the prescribed form for coding instructions, so that $\{a\}$ is the empty functional. A functional will be called *partial recursive* just in case it is one of the functionals $\{a\}$ $(a \in \omega)$.

Many of the indices will have the same interpretation as before. For example, we shall arrange that

$$\{\langle 1, k, l, b, c_0, \ldots, c_{k'-1}\rangle\}(\mathbf{m}, \boldsymbol{\alpha}) \simeq \{b\}(\{c_0\}(\mathbf{m}, \boldsymbol{\alpha}), \ldots, \{c_{k'-1}\}(\mathbf{m}, \boldsymbol{\alpha}), \boldsymbol{\alpha})$$

so that the partial recursive functionals are closed under composition. The key to obtaining the full self-determination principle is the fact that we shall have

$$\{\langle 2, k+1, l\rangle\}(b, \mathbf{m}, \boldsymbol{\alpha}) \simeq \{b\}(\mathbf{m}, \boldsymbol{\alpha}).$$

The index $\langle 2, k+1, l\rangle$ may be thought of as coding the following program: apply the program coded by the first number argument to the remaining arguments. If $\{b\}(\mathbf{m}, \boldsymbol{\alpha})$ is undefined, so is $\{\langle 2, k+1, l\rangle\}(b, \mathbf{m}, \boldsymbol{\alpha})$; it is for this reason that we are forced to deal with partial functionals. It is now evident that the full self-determination principle holds: if G is partial recursive, so is the functional F defined by:

$$\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}) \simeq \{\mathsf{G}(\mathbf{m}, \boldsymbol{\alpha})\}(\mathbf{m}, \boldsymbol{\alpha}) \simeq \{\langle 2, k+1, l\rangle\}(\mathsf{G}(\mathbf{m}, \boldsymbol{\alpha}), \mathbf{m}, \boldsymbol{\alpha}).$$

The reader may well be puzzled as to why a diagonal argument does not exhibit an inconsistency. Indeed, if we set $\mathsf{F}(a, \mathbf{m}, \boldsymbol{\alpha}) \simeq \{a\}(a, \mathbf{m}, \boldsymbol{\alpha}) + 1$, then F is partial recursive and thus coincides with some $\{b\}$. Thus

$$\mathsf{F}(b, \mathbf{m}, \boldsymbol{\alpha}) \simeq \{b\}(b, \mathbf{m}, \boldsymbol{\alpha}) + 1 \simeq \mathsf{F}(b, \mathbf{m}, \boldsymbol{\alpha}) + 1,$$

but F is now a *partial* functional and from this we conclude merely that $\mathsf{F}(b, \mathbf{m}, \boldsymbol{\alpha})$ is not defined.

To realize these provisions, we shall define inductively the set Ω of sequences $(a, \mathbf{m}, \boldsymbol{\alpha}, n)$ such that $\{a\}(\mathbf{m}, \boldsymbol{\alpha}) \simeq n$. At each stage $r+1$ we add to Ω^{r+1} sequences corresponding to computations which are intuitively one step more complex than computations corresponding to sequences in Ω^r. The elements of Ω^0 correspond to computations of some initial functionals which we regard as being immediately computable. These are the same as the initial primitive recursive functionals with two additions which serve as a technical device to enable us to derive the closure of the class of partial recursive functionals under primitive recursion without putting this into the definition. The function Sb_0 is defined in Lemma 2.5 below.

2.1 Definition. Ω is the smallest set such that for all k, l, n, p, q, r, and s, all $i < k$ and $j < l$ and all $(\mathbf{m}, \boldsymbol{\alpha}) \in {}^{k,l}\omega$,

(0) $(\langle 0, k, l, 0, n\rangle, \mathbf{m}, \boldsymbol{\alpha}, n) \in \Omega$;
$(\langle 0, k, l, 1, i\rangle, \mathbf{m}, \boldsymbol{\alpha}, m_i) \in \Omega$;
$(\langle 0, k, l, 2, i\rangle, \mathbf{m}, \boldsymbol{\alpha}, m_i + 1) \in \Omega$;
$(\langle 0, k, l, 3, i, j\rangle, \mathbf{m}, \boldsymbol{\alpha}, \alpha_j(m_i)) \in \Omega$;
$(\langle 0, k+4, l, 4\rangle, p, q, r, s, \mathbf{m}, \boldsymbol{\alpha}, p) \in \Omega$, if $r = s$;
$(\langle 0, k+4, l, 4\rangle, p, q, r, s, \mathbf{m}, \boldsymbol{\alpha}, q) \in \Omega$, if $r \neq s$;
$(\langle 0, k+2, l, 5\rangle, p, q, \mathbf{m}, \boldsymbol{\alpha}, \mathrm{Sb}_0(p, q)) \in \Omega$;

(1) for any k', b, $c_0, \ldots, c_{k'-1}$, $q_0, \ldots, q_{k'-1}$, if for all $i < k'$ $(c_i, \mathbf{m}, \boldsymbol{\alpha}, q_i) \in \Omega$ and $(b, \mathbf{q}, \boldsymbol{\alpha}, n) \in \Omega$, then

$$(\langle 1, k, l, b, c_0, \ldots, c_{k'-1}\rangle, \mathbf{m}, \boldsymbol{\alpha}, n) \in \Omega;$$

(2) for any b, if $(b, \mathbf{m}, \boldsymbol{\alpha}, n) \in \Omega$, then

$$(\langle 2, k+1, l\rangle, b, \mathbf{m}, \boldsymbol{\alpha}, n) \in \Omega.$$

Although it is not phrased explicitly as such, the definition of Ω is another example of closure under finitary functions. For example, clause (2) requires Ω to be closed under the function $\lambda x.(\langle 2, k+1, l\rangle) * x$. In particular, there is an inductive operator Γ such that $\Omega = \bar{\Gamma} = \Gamma^{(\omega)}$.

2.2 Lemma. *For all a, $\mathbf{m}$, and $\boldsymbol{\alpha}$, there is at most one n such that $(a, \mathbf{m}, \boldsymbol{\alpha}, n) \in \Omega$.*

Proof. With Γ as above, it suffices to prove by induction on r that there is at most one n such that $(a, \mathbf{m}, \boldsymbol{\alpha}, n) \in \Gamma^r$. Assume as induction hypothesis that this holds for all $s < r$ and suppose that both $(a, \mathbf{m}, \boldsymbol{\alpha}, n)$ and $(a, \mathbf{m}, \boldsymbol{\alpha}, n')$, with $n \neq n'$, belong to Γ^r. This is clearly impossible if $(a)_0 = 0$ so suppose $(a)_0 = 1$. Then for some k, k', l, b, c, $\mathbf{q}$, and $\mathbf{q}'$, $a = \langle 1, k, l, b, c\rangle$, and for all $i < k'$, $(c_i, \mathbf{m}, \boldsymbol{\alpha}, q_i)$ and $(c_i, \mathbf{m}, \boldsymbol{\alpha}, q'_i)$ belong to $\Gamma^{(r)}$ and $(b, \mathbf{q}, \boldsymbol{\alpha}, n)$ and $(b, \mathbf{q}', \boldsymbol{\alpha}, n')$ belong to $\Gamma^{(r)}$. But then by the induction hypothesis, $\mathbf{q} = \mathbf{q}'$ so $n = n'$. The case $(a)_0 = 2$ is similar. $\square$

Thus if we set

$$\{a\}(\mathbf{m}, \boldsymbol{\alpha}) \simeq n \leftrightarrow (a, \mathbf{m}, \boldsymbol{\alpha}, n) \in \Omega,$$

then for each $a \in \omega$, $\{a\}$ is a partial functional.

2.3 Definition. For all F and R,

(i) F is *partial recursive* iff for some $a \in \omega$, $\mathsf{F} = \{a\}$; a is called an *index for* F;

(ii) F is *recursive* iff F is partial recursive and total;

(iii) R is *recursive* iff K_{R} is recursive; an index for K_{R} is also called an *index for* R.

2.4 Remark. There is a somewhat technical point which should be mentioned here. The assertion

$$\{\langle 2, k+1, l\rangle\}(b, \mathbf{m}, \boldsymbol{\alpha}) \simeq \{b\}(\mathbf{m}, \boldsymbol{\alpha})$$

is equivalent to: for all n,

$$\{\langle 2, k+1, l\rangle\}(b, \mathbf{m}, \boldsymbol{\alpha}) \simeq n \leftrightarrow \{b\}(\mathbf{m}, \boldsymbol{\alpha}) \simeq n.$$

The implication $(\leftarrow)$ is built into the definition, but the converse must be proved. The situation is similar with

$$\{\langle 1, k, l, b, c_0, \ldots, c_{k'-1}\rangle\}(\mathbf{m}, \boldsymbol{\alpha}) \simeq \{b\}(\{c_0\}(\mathbf{m}, \boldsymbol{\alpha}), \ldots, \{c_{k'-1}\}(\mathbf{m}, \boldsymbol{\alpha}), \boldsymbol{\alpha}).$$

If the right-hand side is $\simeq n$, then for some $q_0, \ldots, q_{k'-1}$, each $\{c_i\}(\mathbf{m}, \boldsymbol{\alpha}) \simeq q_i$ and $\{b\}(\mathbf{q}, \boldsymbol{\alpha}) \simeq n$ so that also the left-hand side $\simeq n$ by the definition of Ω. The reader should not find great difficulty in supplying the missing proofs. (Exercise 2.14)

From this discussion it is clear that the class of partial recursive functionals is closed under functional composition. To show that it is also closed under primitive recursion, we first prove a very fundamental result known as the Recursion Theorem.

2.5 Lemma. *For every $i \in \omega$, there exists a primitive recursive function* Sb_i *such that for any a, any $k > i$, any l, and any $(\mathbf{m}, \boldsymbol{\alpha}) \in {}^{k,l}\omega$,*

$$\{\mathrm{Sb}_i(a, m_0, \ldots, m_i)\}(m_{i+1}, \ldots, m_{k-1}, \boldsymbol{\alpha}) \simeq \{a\}(\mathbf{m}, \boldsymbol{\alpha}).$$

Proof. Let $k' = (a)_1 - 1$ and $l' = (a)_2$. Then set

$$\mathrm{Sb}_0(a, m_0) = \langle 1, k', l', a, \langle 0, k', l', 0, m_0\rangle, \langle 0, k', l', 1, 0\rangle, \ldots, \langle 0, k', l', 1, k'-1\rangle\rangle;$$

$$\mathrm{Sb}_{i+1}(a, m_0, \ldots, m_{i+1}) = \mathrm{Sb}_0(\mathrm{Sb}_i(a, m_0, \ldots, m_i), m_{i+1}).$$

The functions Sb_i are clearly all primitive recursive. We leave it to the reader to check that they all have the required property. □

2.6 Recursion Theorem. *For any partial recursive* F *there exists an $\bar{e} \in \omega$ such that for all $(\mathbf{m}, \boldsymbol{\alpha})$,*

$$\{\bar{e}\}(\mathbf{m}, \boldsymbol{\alpha}) \simeq \mathsf{F}(\bar{e}, \mathbf{m}, \boldsymbol{\alpha}).$$

Proof. Let F be given and set $\mathsf{G}(a, \mathbf{m}, \boldsymbol{\alpha}) \simeq \mathsf{F}(\mathrm{Sb}_0(a, a), \mathbf{m}, \boldsymbol{\alpha})$. G is partial recursive, so let b be an index for G and take $\bar{e} = \mathrm{Sb}_0(b, b)$. Then

$$\{\bar{e}\}(\mathbf{m}, \boldsymbol{\alpha}) \simeq \{b\}(b, \mathbf{m}, \boldsymbol{\alpha}) \simeq \mathsf{G}(b, \mathbf{m}, \boldsymbol{\alpha}) \simeq \mathsf{F}(\mathrm{Sb}_0(b, b), \mathbf{m}, \boldsymbol{\alpha})$$
$$\simeq \mathsf{F}(\bar{e}, \mathbf{m}, \boldsymbol{\alpha}). \quad \square$$

We observe that the Recursion Theorem holds also for primitive recursion. If F is primitive recursive, then so is G in the preceding proof. If b is a primitive recursive index for G, then $\bar{e} = \mathrm{Sb}_0(b, b)$ is also a primitive recursive index and clearly

$$[\bar{e}](m, \alpha) = \mathsf{F}(\bar{e}, m, \alpha).$$

We shall refer to this fact as the *Primitive Recursion Theorem.*

The following is a property of the partial recursive functionals which is not shared by the primitive recursive functionals.

2.7 Definition (*unbounded search*). For any functional G of rank $(k+1, l)$, $\mathrm{Se}(\mathsf{G})$ is the functional F of rank (k, l) such that

$$\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}) \simeq \begin{cases} q, \text{ if } \mathsf{G}(q, \mathbf{m}, \boldsymbol{\alpha}) \simeq 0, \text{ but for all } r < q, \\ \quad \text{there exists } n > 0 \text{ such that } \mathsf{G}(r, \mathbf{m}, \boldsymbol{\alpha}) \simeq n; \\ \\ \text{undefined, if there is no such } q. \end{cases}$$

(We write $\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}) \simeq$ "least" $q \,.\, \mathsf{G}(q, \mathbf{m}, \boldsymbol{\alpha}) \simeq 0$. Note that if $\mathsf{G}(r, \mathbf{m}, \boldsymbol{\alpha})$ is undefined for some r less than the smallest q such that $\mathsf{G}(q, \mathbf{m}, \boldsymbol{\alpha}) \simeq 0$ (if any), then $\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha})$ is undefined. If this cannot happen, say, because G is total, we write $\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}) \simeq \text{least } q \,.\, \mathsf{G}(q, \mathbf{m}, \boldsymbol{\alpha}) \simeq 0$. In particular, we write least $p \,.\, \mathsf{R}(p, \mathbf{m}, \boldsymbol{\alpha})$ for least $p \,.\, \mathsf{K}_{\mathsf{R}}(p, \mathbf{m}, \boldsymbol{\alpha}) = 0$.)

2.8 Theorem. *The class of partial recursive functionals is closed under unbounded search.*

Proof. Suppose G is partial recursive and $\mathsf{F} = \mathrm{Se}(\mathsf{G})$. Let b and c be indices such that for all $e, q, \mathbf{m}$, and $\boldsymbol{\alpha}$,

$$\{b\}(e, q, \mathbf{m}, \boldsymbol{\alpha}) \simeq 0, \quad \text{and} \quad \{c\}(e, q, \mathbf{m}, \boldsymbol{\alpha}) \simeq \{e\}(q+1, \mathbf{m}, \boldsymbol{\alpha}) + 1,$$

and set

$$\mathsf{H}(e, q, \mathbf{m}, \boldsymbol{\alpha}) \simeq \{\{\langle 0, 4, 0, 4 \rangle\}(b, c, \mathsf{G}(q, \mathbf{m}, \boldsymbol{\alpha}), 0)\}(e, q, \mathbf{m}, \boldsymbol{\alpha}).$$

By the Recursion Theorem there exists an $\bar{e}$ such that

$$\{\bar{e}\}(q, \mathbf{m}, \boldsymbol{\alpha}) \simeq \mathsf{H}(\bar{e}, q, \mathbf{m}, \boldsymbol{\alpha}).$$

Let $\mathsf{I} = \{\bar{e}\}$.

Suppose first that $\mathsf{G}(q, \mathbf{m}, \boldsymbol{\alpha}) \simeq 0$. Then

$$\{\langle 0, 4, 0, 4 \rangle\}(b, c, \mathsf{G}(q, \mathbf{m}, \boldsymbol{\alpha}), 0) \simeq b$$

so

$$\mathsf{I}(q, \mathbf{m}, \boldsymbol{\alpha}) \simeq \mathsf{H}(\bar{e}, q, \mathbf{m}, \boldsymbol{\alpha}) \simeq \{b\}(\bar{e}, q, \mathbf{m}, \boldsymbol{\alpha}) \simeq 0.$$

On the other hand, if $\mathsf{G}(q, \mathbf{m}, \boldsymbol{\alpha}) \simeq n > 0$, then

$$\mathsf{I}(q, \mathbf{m}, \boldsymbol{\alpha}) \simeq \{c\}(\bar{e}, q, \mathbf{m}, \boldsymbol{\alpha}) \simeq \mathsf{I}(q+1, \mathbf{m}, \boldsymbol{\alpha}) + 1.$$

If $\mathsf{G}(q, \mathbf{m}, \boldsymbol{\alpha})$ is undefined, so is $\mathsf{I}(q, \mathbf{m}, \boldsymbol{\alpha})$. From these facts it follows easily that $\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}) \simeq \mathsf{I}(0, \mathbf{m}, \boldsymbol{\alpha})$. $\square$

2.9 Corollary. *The predecessor function $p \dot{-} 1$ is partial recursive.*

Proof. Let b and c be indices such that for all p,

$$\{b\}(p) \simeq 0 \quad \text{and} \quad \{c\}(p) \simeq \text{“least”}\, q . \{\langle 0,4,0,4\rangle\}(0,1,q+1,p) \simeq 0.$$

Then

$$p \dot{-} 1 \simeq \{\{\langle 0,4,0,4\rangle\}(b,c,p,0)\}(p). \quad \square$$

2.10 Theorem. *The class of partial recursive functionals is closed under primitive recursion and course-of-values recursion.*

Proof. Suppose G and H are partial recursive and F = Rec(G, H). Let G′ and H′ be defined by:

$$\mathsf{G}'(e,p,\mathbf{m},\boldsymbol{\alpha}) \simeq \mathsf{G}(\mathbf{m},\boldsymbol{\alpha});$$

$$\mathsf{H}'(e,p,\mathbf{m},\boldsymbol{\alpha}) \simeq \mathsf{H}(\{e\}(p \dot{-} 1,\mathbf{m},\boldsymbol{\alpha}), p \dot{-} 1,\mathbf{m},\boldsymbol{\alpha}).$$

By the corollary, G′ and H′ are partial recursive. Let b' and c' be indices for G′ and H′, respectively, and set

$$\mathsf{I}(e,p,\mathbf{m},\boldsymbol{\alpha}) \simeq \{\{\langle 0,4,0,4\rangle\}(b',c',p,0)\}(e,p,\mathbf{m},\boldsymbol{\alpha}).$$

Thus

$$\mathsf{I}(e,0,\mathbf{m},\boldsymbol{\alpha}) \simeq \mathsf{G}(\mathbf{m},\boldsymbol{\alpha})$$

and

$$\mathsf{I}(e,p+1,\mathbf{m},\boldsymbol{\alpha}) \simeq \mathsf{H}(\{e\}(p,\mathbf{m},\boldsymbol{\alpha}),p,\mathbf{m},\boldsymbol{\alpha}).$$

Let $\bar{e}$ be an index provided by the Recursion Theorem such that

$$\{\bar{e}\}(p,\mathbf{m},\boldsymbol{\alpha}) \simeq \mathsf{I}(\bar{e},p,\mathbf{m},\boldsymbol{\alpha}).$$

Then

$$\{\bar{e}\}(0,\mathbf{m},\boldsymbol{\alpha}) \simeq \mathsf{G}(\mathbf{m},\boldsymbol{\alpha})$$

and

$$\{\bar{e}\}(p+1,\mathbf{m},\boldsymbol{\alpha}) \simeq \mathsf{H}(\{\bar{e}\}(p,\mathbf{m},\boldsymbol{\alpha}),p,\mathbf{m},\boldsymbol{\alpha}).$$

So $\mathsf{F} = \{\bar{e}\}$ and thus F is partial recursive. □

2.11 Corollary. *Every primitive recursive functional is recursive.* □

We leave it to the reader to check the obvious fact that the primitive recursive functionals form a proper subset of the recursive functionals (Exercise 2.17).

Computations of partial recursive functionals may be arranged in labeled trees similar to those for primitive recursions. There is one additional type of node:

$$\begin{array}{l} \vdots \\ \bullet\ (\langle 2, k+1, l\rangle, b, \mathbf{m}, \boldsymbol{\alpha}) \\ | \\ \bullet\ (b, \mathbf{m}, \boldsymbol{\alpha}) \\ \vdots \end{array}$$

Branching in the tree is still finite, so that a tree either is finite or has an infinite branch. The latter case corresponds to an undefined computation. The procedure for computing on the tree is the same as before and we conclude that the functional $\lambda\, a\, \mathbf{m}\, \boldsymbol{\alpha} . \{a\}(\mathbf{m}, \boldsymbol{\alpha})$ is mechanically computable, and hence so is every partial recursive functional.

We want now to consider other closure conditions satisfied by the (partial) recursive functionals and the recursive relations. It follows as in 1.3 that the recursive relations form a Boolean algebra. If we restrict ourselves to recursive (total) functionals, then we can establish closure under bounded search, definition by cases, relational composition, and bounded quantification with exactly the same proofs as for Theorem 1.5. The proof for expansion given there depends on the inductive characterization of the class Prf and does not generalize directly; the result is proved as Corollary 3.4.

What happens to these properties when the functionals are allowed to be partial? Closure under relational composition is no longer true (Exercise 4.26). On the other hand it is easy to verify that the proof for bounded search still works: if G is partial recursive so is $\mathrm{Bs}(\mathsf{G})$. The proof for definition by cases, however, does not work if some of $\mathsf{G}_0, \ldots, \mathsf{G}_{k'}$ are not total functionals. If F is as defined in 1.5(iii), $\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha})$ is undefined unless *all* of $\mathsf{G}_0(\mathbf{m}, \boldsymbol{\alpha}), \ldots, \mathsf{G}_{k'}(\mathbf{m}, \boldsymbol{\alpha})$ are defined, whereas $\mathrm{Cases}_{k'}(\mathsf{G}, \mathsf{R})$ is defined for any $(\mathbf{m}, \boldsymbol{\alpha})$ such that for some i, $\mathsf{R}_i(\mathbf{m}, \boldsymbol{\alpha})$ and $\mathsf{G}_i(\mathbf{m}, \boldsymbol{\alpha})$ is defined. The result is nevertheless true.

2.12 Theorem. *The class of partial recursive functionals and recursive relations is closed under definition by cases.*

Proof. Suppose $\mathsf{R}_0, \ldots, \mathsf{R}_{k'-1}$ are recursive relations and $\mathsf{G}_0, \ldots, \mathsf{G}_{k'}$ are partial recursive functionals which satisfy the hypothesis of 1.4(iii). Let b_i be an index for G_i and let H be the recursive functional defined by:

$$\mathsf{H}(\mathbf{m}, \boldsymbol{\alpha}) = \begin{cases} b_0, & \text{if } \mathsf{R}_0(\mathbf{m}, \boldsymbol{\alpha}); \\ \quad \vdots & \\ b_{k'-1}, & \text{if } \mathsf{R}_{k'-1}(\mathbf{m}, \boldsymbol{\alpha}); \\ b_{k'}, & \text{otherwise.} \end{cases}$$

Then $\mathrm{Cases}_{k'}(\mathsf{G}_0, \ldots, \mathsf{G}_{k'}, \mathsf{R}_0, \ldots, \mathsf{R}_{k'-1})$ is the partial recursive functional

$$\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}) \simeq \{\mathsf{H}(\mathbf{m}, \boldsymbol{\alpha})\}(\mathbf{m}, \boldsymbol{\alpha}). \quad \square$$

2.13–2.20 Exercises

2.13. Show that there exists a primitive recursive functional F such that if for each $\boldsymbol{\alpha}$ we define

$$\Gamma_{\boldsymbol{\alpha}}(A) = \{u : \exists p\, (\forall i < p)\, .\, \mathsf{F}(p, i, u, \langle \boldsymbol{\alpha} \rangle) \in A \cup \{0\}\},$$

then

$$\{a\}(\mathbf{m}, \boldsymbol{\alpha}) \simeq n \leftrightarrow \langle a, \mathbf{m}, n \rangle \in \bar{\Gamma}_{\boldsymbol{\alpha}}.$$

2.14. Supply the missing proofs discussed in Remark 2.4.

2.15. Prove the *Effective Recursion Theorem*: there exists a primitive recursive function g such that for all $a, \mathbf{m}$, and $\boldsymbol{\alpha}$,

$$\{g(a)\}(\mathbf{m}, \boldsymbol{\alpha}) \simeq \{a\}(g(a), \mathbf{m}, \boldsymbol{\alpha}).$$

2.16. With each sequence $(a, \mathbf{m}, \boldsymbol{\alpha}, n)$ such that $\{a\}(\mathbf{m}, \boldsymbol{\alpha}) \simeq n$, we associate the natural number $|a, \mathbf{m}, \boldsymbol{\alpha}|$, the least number r such that $(a, \mathbf{m}, \boldsymbol{\alpha}, n) \in \Omega^r$. Show that $\bar{e}$ in the Recursion Theorem may be chosen so that whenever $\mathsf{F}(\bar{e}, \mathbf{m}, \boldsymbol{\alpha}) \simeq n$, then $|a, \bar{e}, \mathbf{m}, \boldsymbol{\alpha}| < |\bar{e}, \mathbf{m}, \boldsymbol{\alpha}|$ (a is an index for F).

2.17. Check that the class of primitive recursive functionals is a proper subclass of the class of recursive functionals.

2.18. Prove an effective version of Corollary 2.11: there exists a primitive recursive function f such that for all $a \in \mathrm{Pri}$, $[a] = \{f(a)\}$.

2.19. Suppose that F is defined from G and H by the following recursion:

$$\mathsf{F}(0, q, \mathbf{m}, \boldsymbol{\alpha}) \simeq \mathsf{G}(q, \mathbf{m}, \boldsymbol{\alpha}),$$

$$\mathsf{F}(p+1, q, \mathbf{m}, \boldsymbol{\alpha}) \simeq \mathsf{H}(p, q, \mathbf{m}, \boldsymbol{\alpha}, \lambda r\, .\, \mathsf{F}(p, r, \mathbf{m}, \boldsymbol{\alpha})).$$

If G and H are partial recursive, does it follow that F is partial recursive?

2.20. Is it possible to define recursive functions Sb_j^1 such that for all a, all $l > j, k$, and $(\mathbf{m}, \boldsymbol{\alpha}) \in {}^{k,l}\omega$,

$$\{\mathrm{Sb}_j^1(a, \alpha_0, \ldots, \alpha_j)\}(\mathbf{m}, \alpha_{j+1}, \ldots, \alpha_{l-1}) \simeq \{a\}(\mathbf{m}, \boldsymbol{\alpha})?$$

2.21 Notes. The definition we have given for the class of partial recursive functionals via indexed schemata is an adaptation of that of Kleene [1959]. Historically, the most important characterizations of this class are those of Kleene [1935/36] and Turing [1936]. A good informal discussion of these may be found in Rogers [1967, § 1.5]. We have chosen the Kleene [1959] definition for two reasons. First the basic properties of the partial recursive functionals are easier to derive because the indexing is built in from the beginning. More importantly, it is this definition which is most easily generalized to functionals of higher types and functions on ordinals. Of course, there are many minor variations possible. For example, the schemata in clause (0) for equality and sequence coding may be omitted in favor of a schema for primitive recursion. The sequence coding functions may be replaced by $+$ and $\cdot$.

The fact that recursiveness for functionals can be defined exactly as for functions was apparently first noted in print in Kleene [1952, § 55] and developed in Kleene [1955 b].

3. *Normal Forms*

Much of the remaining theory of partial recursive functionals depends on the fact that the relation $\{a\}(\mathbf{m}, \boldsymbol{\alpha}) \simeq n$ can be represented in a simple way in terms of primitive recursive relations, namely, as $\exists u\, \mathsf{P}(u, a, \mathbf{m}, n, \boldsymbol{\alpha})$ with P primitive recursive. The key idea is that the number u codes the sequence of steps which are necessary to carry out the computation.

3.1 Theorem. *There exists a primitive recursive relation $\mathsf{T} \subseteq {}^{3,1}\omega$ such that for all $a, \mathbf{m}, \boldsymbol{\alpha}$, and n,*

$$\{a\}(\mathbf{m}, \boldsymbol{\alpha}) \simeq n \leftrightarrow \exists u\, [\mathsf{T}(a, \langle \mathbf{m} \rangle, u, \langle \boldsymbol{\alpha} \rangle) \wedge (u)_0 = n].$$

Proof. We first define a primitive recursive relation Cmt—$\mathsf{Cmt}(u, \langle \boldsymbol{\alpha} \rangle)$ is to mean that u codes some computation relative to $\boldsymbol{\alpha}$. Each such u will be of the form $u = \langle n, a, \mathbf{m}, v_0, \ldots, v_{r-1} \rangle$. $v_0, \ldots, v_{r-1}$ code in turn the immediate subcomputations of u, that is the computations indicated by the labels on the immediate predecessors of the node labeled $(a, \mathbf{m}, \boldsymbol{\alpha})$ in a computation tree. We shall write $u = \langle n, a, \mathbf{m}, \ldots \rangle$ as an abbreviation for: $\mathrm{Sq}(u)$ and $\lg(u) \geq (a)_1 + 2$ and $(u)_0 = n$ and $(u)_1 = a$ and $(\forall i < (a)_1)(u)_{i+2} = m_i$.

For any l and any $\alpha \in {}^{l}({}^{\omega}\omega)$, $\mathsf{Cmt}(u, \langle \alpha \rangle)$ iff for some $k, m_0, \ldots, m_{k-1}, n, p, q, r$, and s all less than u, one of the following holds:

(0) $u = \langle n, \langle 0, k, l, 0, n \rangle, \mathbf{m} \rangle$;
$u = \langle m_i, \langle 0, k, l, 1, i \rangle, \mathbf{m} \rangle$ for some $i < k$;
$u = \langle m_i + 1, \langle 0, k, l, 2, i \rangle, \mathbf{m} \rangle$ for some $i < k$;
$u = \langle \alpha_j(m_i), \langle 0, k, l, 3, i, j \rangle, \mathbf{m} \rangle$ for some $i < k$ and $j < l$;
$u = \langle p, \langle 0, k+4, l, 4 \rangle, p, q, r, s, \mathbf{m} \rangle$ and $r = s$;
$u = \langle q, \langle 0, k+4, l, 4 \rangle, p, q, r, s, \mathbf{m} \rangle$ and $r \neq s$;
$u = \langle \mathsf{Sb}_0(p, q), \langle 0, k+2, l, 5 \rangle, p, q, \mathbf{m} \rangle$;

(1) for some $k', b, c_0, \ldots, c_{k'-1}, q_0, \ldots, q_{k'-1}, v_0, \ldots, v_{k'} < u$,

$$u = \langle n, \langle 1, k, l, b, \mathbf{c} \rangle, \mathbf{m}, \mathbf{v} \rangle, \text{ for all } i \leq k', \ \mathsf{Cmt}(v_i, \langle \alpha \rangle),$$

$$\text{for all } i < k', \ v_i = \langle q_i, c_i, \mathbf{m}, \ldots \rangle, \text{ and } v_{k'} = \langle n, b, \mathbf{q}, \ldots \rangle;$$

(2) for some b and $v < u$,

$$u = \langle n, \langle 2, k+1, l \rangle, b, \mathbf{m}, v \rangle, \quad \mathsf{Cmt}(v, \langle \alpha \rangle), \quad \text{and} \quad v = \langle n, b, \mathbf{m}, \ldots \rangle.$$

All quantifiers in the definition of Cmt are bounded to u and thus Cmt may be seen to be primitive recursive by the techniques discussed following Theorem 1.8. We now set

$$\mathsf{T}(a, \langle \mathbf{m} \rangle, u, \langle \alpha \rangle) \leftrightarrow \mathsf{Cmt}(u, \langle \alpha \rangle) \wedge u = \langle (u)_0, a, \mathbf{m}, \ldots \rangle.$$

Then T is clearly primitive recursive and it suffices to show that for all $a, \mathbf{m}, \alpha$, and n,

(i) $\{a\}(\mathbf{m}, \alpha) \simeq n \rightarrow \exists u\, [\mathsf{T}(a, \langle \mathbf{m} \rangle, u, \langle \alpha \rangle) \wedge (u)_0 = n]$;

(ii) $\forall u\, [\mathsf{T}(a, \langle \mathbf{m} \rangle, u, \langle \alpha \rangle) \rightarrow \{a\}(\mathbf{m}, \alpha) \simeq (u)_0]$.

The proof of (i) is by induction over Ω — that is, we show that the set

$$Z = \{(a, \mathbf{m}, \alpha, n) : \exists u\, [\mathsf{T}(a, \langle \mathbf{m} \rangle, u, \langle \alpha \rangle) \wedge (u)_0 = n]\}$$

is closed under the conditions which define Ω. If $(a)_0 = 0$ and $(a, \mathbf{m}, \alpha, n) \in \Omega$, then the condition for membership in Z is satisfied with $u = \langle n, a, \mathbf{m} \rangle$. Suppose $a = \langle 1, k, l, b, \mathbf{c} \rangle$ and for some $\mathbf{q}$, $(\forall i < k')(c_i, \mathbf{m}, \alpha, q_i) \in Z$ and $(b, \mathbf{q}, \alpha, n) \in Z$. Then there exist $v_0, \ldots, v_{k'}$ such that $(\forall i < k')[\mathsf{T}(c_i, \langle \mathbf{m} \rangle, v_i, \langle \alpha \rangle) \wedge (v_i)_0 = q_i]$ and $[\mathsf{T}(b, \langle \mathbf{q} \rangle, v_{k'}, \langle \alpha \rangle) \wedge (v_{k'})_0 = n]$. Then by clause (1) in the definition of Cmt, if $u = \langle n, a, \mathbf{m}, \mathbf{v} \rangle$, $\mathsf{Cmt}(u, \langle \alpha \rangle)$ and hence $\mathsf{T}(a, \langle \mathbf{m} \rangle, u, \langle \alpha \rangle)$. Since also $(u)_0 = n$, $(a, \mathbf{m}, \alpha, n) \in Z$. The proof for clause (2) is easier.

The proof of (ii) is by ordinary induction on u. We assume as induction hypothesis that the matrix of (ii) is true for all $v < u$ and that $\mathsf{T}(a, \langle \mathbf{m} \rangle, u, \langle \alpha \rangle)$. If u falls under clause (0), the result is obvious. Suppose u falls under clause (1). Then since $v_0, \ldots, v_{k'} < u$, and clearly $\mathsf{T}(c_i, \langle \mathbf{m} \rangle, v_i, \langle \alpha \rangle)$ and $T(b, \langle \mathbf{q} \rangle, v_{k'}, \langle \alpha \rangle)$, the induction hypothesis guarantees that $\{c_i\}(\mathbf{m}, \boldsymbol{\alpha}) \simeq q_i$ and $\{b\}(\mathbf{q}, \boldsymbol{\alpha}) \simeq n$. Hence $\{a\}(\mathbf{m}, \boldsymbol{\alpha}) \simeq n = (u)_0$. Case (2) is similar. □

3.2 Corollary (First Normal Form). *For any partial recursive functional* F, *there exists an a such that*

$$\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}) \simeq (\text{least } u \,.\, \mathsf{T}(a, \langle \mathbf{m} \rangle, u, \langle \boldsymbol{\alpha} \rangle))_0.$$

Proof. If a is any index for F, then clearly $\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha})$ is defined just in case $\exists u\, \mathsf{T}(a, \langle \mathbf{m} \rangle, u, \langle \boldsymbol{\alpha} \rangle)$ and if so, then $\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}) \simeq (u)_0$ for any such u. □

As an immediate consequence of the First Normal Form we obtain direct inductive characterizations of the classes of partial recursive and recursive functionals:

3.3 Corollary. (i) *The class of partial recursive functionals is the smallest class of partial functionals which includes the class of primitive recursive functionals and is closed under functional composition and unbounded search*;

(ii) *the class of recursive functionals is the smallest class of total functionals which includes the class of primitive recursive functionals and is closed under functional composition and unbounded search restricted to functionals* G *such that* $\forall \mathbf{m} \forall \boldsymbol{\alpha}\, \exists q \,.\, \mathsf{G}(q, \mathbf{m}, \boldsymbol{\alpha}) = 0$.

Proof. Let X be the class of partial recursive functionals and Y the other class described in (i). Then $Y \subseteq X$ by the results of § 2 and $X \subseteq Y$ by Corollary 3.2. The proof of (ii) is similar. □

3.4 Corollary. *The class of partial recursive functionals is closed under expansion.*

Proof. Suppose $\mathsf{F}(\mathbf{m}, \mathbf{n}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \simeq \mathsf{G}(\mathbf{m}, \boldsymbol{\alpha})$ and let b be an index for G. By expansion for primitive recursive functionals there is a primitive recursive relation R such that

$$\mathsf{R}(u, \mathbf{m}, \mathbf{n}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \leftrightarrow \mathsf{T}(b, \langle \mathbf{m} \rangle, u, \langle \boldsymbol{\alpha} \rangle).$$

Then

$$\mathsf{F}(\mathbf{m}, \mathbf{n}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \simeq (\text{least } u \,.\, \mathsf{R}(u, \mathbf{m}, \mathbf{n}, \boldsymbol{\alpha}, \boldsymbol{\beta}))_0$$

and is thus partial recursive by Theorem 2.11. □

In the introduction to this chapter we observed that all mechanically calculable functionals are continuous. The fact that all partial recursive functionals are partial continuous plays an important role in the theory of the later chapters and we shall find useful a second normal form which makes this explicit.

3.5 Theorem. *There exists a primitive recursive relation $T \subseteq {}^4\omega$ such that for all $a, \mathbf{m}, \boldsymbol{\alpha}$, and n,*

$$\{a\}(\mathbf{m}, \boldsymbol{\alpha}) \simeq n \leftrightarrow \exists u\,[T(a, \langle\mathbf{m}\rangle, u, \langle\bar{\alpha}_0(u), \ldots, \bar{\alpha}_{l-1}(u)\rangle) \wedge (u)_0 = n].$$

Proof. Let Cmt be the relation defined in the proof of Theorem 3.1 and set

$$\mathrm{Cmt}(u, s) \leftrightarrow \exists \boldsymbol{\beta}\,[s = \langle\bar{\beta}_0(u), \ldots, \bar{\beta}_{l-1}(u)\rangle \wedge \mathsf{Cmt}(u, \langle\boldsymbol{\beta}\rangle)],$$

and

$$T(a, \langle\mathbf{m}\rangle, u, s) \leftrightarrow \mathrm{Cmt}(u, s) \wedge u = \langle(u)_0, a, \mathbf{m}, \ldots\rangle.$$

Suppose first that $\{a\}(\mathbf{m}, \boldsymbol{\alpha}) \simeq n$ and by Theorem 3.1 choose u such that $\mathsf{T}(a, \langle\mathbf{m}\rangle, u, \langle\boldsymbol{\alpha}\rangle)$ and $(u)_0 = n$. Since $\mathsf{Cmt}(u, \langle\boldsymbol{\alpha}\rangle)$ holds we have $\mathrm{Cmt}(u, \langle\bar{\alpha}_0(u), \ldots, \bar{\alpha}_{l-1}(u)\rangle)$ and thus $T(a, \langle\mathbf{m}\rangle, u, \langle\bar{\alpha}_0(u), \ldots, \bar{\alpha}_{l-1}(u)\rangle)$. For the implication ($\leftarrow$) we first note that it is easy to prove by induction on u that

$$\mathsf{Cmt}(u, \langle\boldsymbol{\alpha}\rangle) \wedge (\forall j < l)\bar{\alpha}_j(u) = \bar{\beta}_j(u) \rightarrow \mathsf{Cmt}(u, \langle\boldsymbol{\beta}\rangle).$$

Intuitively, the computation u "uses" values of the functions only for arguments less than u. Then if $T(a, \langle\mathbf{m}\rangle, u, \langle\bar{\alpha}_0(u), \ldots, \bar{\alpha}_{l-1}(u)\rangle)$ holds, it follows that also $\mathsf{T}(a, \langle\mathbf{m}\rangle, u, \langle\boldsymbol{\alpha}\rangle)$ and the result is immediate from Theorem 3.1.

Of course, as we have defined them above it is not at all evident that Cmt and T are primitive recursive. In fact, however, Cmt may also be defined exactly as was Cmt except for the replacement of the clause

$$u = \langle\alpha_j(m_i), \ldots\rangle \quad \text{by} \quad u = \langle((s)_j)_{m_i}, \ldots\rangle$$

and the addition of the conditions: $\lg(s) = l$ and $(\forall j < l)\lg((s)_j) = u$. $\square$

3.6 Corollary (Second Normal Form). *For any partial recursive functional* F, *there exists an a such that*

$$\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}) \simeq (\text{least } u\,.\, T(a, \langle\mathbf{m}\rangle, u, \langle\bar{\alpha}_0(u), \ldots, \bar{\alpha}_{l-1}(u)\rangle))_0. \quad \square$$

3.7 Corollary. *Every partial recursive functional is partial continuous. Every recursive relation is closed-open.*

Proof. If F is partial recursive and $(\mathbf{m}, \boldsymbol{\alpha}) \in \mathsf{F}^{-1}(\{n\})$, then for some u, $T(a, \langle\mathbf{m}\rangle, u, \langle\bar{\alpha}_0(u), \ldots, \bar{\alpha}_{l-1}(u)\rangle)$ and $(u)_0 = n$. Then for any $(\mathbf{p}, \boldsymbol{\beta})$ belonging to the neighborhood $\{m_0\} \times \cdots \times \{m_{k-1}\} \times [\alpha_0 \restriction u] \times \cdots \times [\alpha_{l-1} \restriction u]$ of $(\mathbf{m}, \boldsymbol{\alpha})$ also $\mathsf{F}(\mathbf{p}, \boldsymbol{\beta}) \simeq n$ so $(\mathbf{p}, \boldsymbol{\beta}) \in \mathsf{F}^{-1}(\{n\})$. Hence $\mathsf{F}^{-1}(\{n\})$ is open and F is partial continuous. If R is recursive, K_R is recursive, hence continuous so R is closed-open by Lemma I.2.1. □

The recursive functionals are sometimes called *effectively continuous* and the recursive relations *effectively closed-open.*

We conclude this section by applying the Second Normal Form to establish another closure property of the class of partial recursive functionals.

3.8 Definition. (i) (*functional substitution*) For any functionals G and H of ranks $(k, l+1)$ and $(k+1, l)$, respectively, FSub(G, H) is the functional F of rank (k, l) such that for all $\mathbf{m}$ and $\boldsymbol{\alpha}$,

$$\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}) \simeq \mathsf{G}(\mathbf{m}, \boldsymbol{\alpha}, \lambda p \,.\, \mathsf{H}(p, \mathbf{m}, \boldsymbol{\alpha}));$$

(ii) (*relational substitution*) for any relation S of rank $(k, l+1)$ and any functional H of rank $(k+1, l)$, RSub(S, H) is the relation R of rank (k, l) such that for all $\mathbf{m}$ and $\boldsymbol{\alpha}$,

$$\mathsf{R}(\mathbf{m}, \boldsymbol{\alpha}) \leftrightarrow \mathsf{S}(\mathbf{m}, \boldsymbol{\alpha}, \lambda p \,.\, \mathsf{H}(p, \mathbf{m}, \boldsymbol{\alpha})).$$

At first consideration, it seems natural that the class of partial recursive functionals should be closed under functional substitution. If G and H are mechanically calculable then so should be F — whenever in the computation of a value $\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha})$ a value of $\lambda p \,.\, \mathsf{H}(p, \mathbf{m}, \boldsymbol{\alpha})$ is called for, it could be computed by a "subroutine". This procedure leads to an answer as long as $\mathsf{H}(p, \mathbf{m}, \boldsymbol{\alpha})$ is defined for all those finitely many p for which a value is required during the computation. The right-hand side, however, is defined at most when $\lambda p \,.\, \mathsf{H}(p, \mathbf{m}, \boldsymbol{\alpha})$ is a *total* function, as we have admitted only total functions as arguments. Thus the procedure described actually calculates an extension of F.

This difficulty could be removed by extending our functionals to allow partial functions as arguments, but this leads to other problems which we prefer to avoid. Of course, this argument establishes only that the proposed procedure may fail to compute F, but in fact the partial recursive functionals are *not* closed under functional substitution (Exercise 4.27). We have instead the following restricted version.

3.9 Theorem. *For any partial recursive functionals* G *and* H *there exists a partial recursive functional* F *such that for all* $\mathbf{m}$ *and* $\boldsymbol{\alpha}$ *such that* $\lambda p \,.\, \mathsf{H}(p, \mathbf{m}, \boldsymbol{\alpha})$ *is total,*

$$\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}) \simeq \mathsf{G}(\mathbf{m}, \boldsymbol{\alpha}, \lambda p \,.\, \mathsf{H}(p, \mathbf{m}, \boldsymbol{\alpha})).$$

Proof. To minimize notational confusion we suppose $k = l = 1$. Let b and c be indices for G and H, respectively. By Theorem 3.5 we have for any m, α, and β,

$$\mathsf{G}(m, \alpha, \beta) \simeq n \leftrightarrow \exists u\, [T(b, \langle m \rangle, u, \langle \bar{\alpha}(u), \bar{\beta}(u) \rangle) \wedge (u)_0 = n]$$
$$\leftrightarrow \exists s\, \exists u\, [s = \bar{\beta}(u) \wedge T(b, \langle m \rangle, u, \langle \bar{\alpha}(u), s \rangle) \wedge (u)_0 = n].$$

Then if $\lambda p\,.\,\mathsf{H}(p, m, \alpha)$ is total,

$$\mathsf{G}(m, \alpha, \lambda p\,.\,\mathsf{H}(p, m, \alpha)) \simeq n \leftrightarrow \exists s\, \exists u\, \exists v\, [(\forall p < u)(\mathsf{T}(c, \langle p, m \rangle, (v)_p, \langle \alpha \rangle)$$
$$\wedge ((v)_p)_0 = (s)_p) \wedge T(b, \langle m \rangle, u, \langle \bar{\alpha}(u), s \rangle) \wedge (u)_0 = n].$$

If we abbreviate the right-hand side of this equivalence by $\exists s\, \exists u\, \exists v$ $\mathsf{P}(s, u, v, m, \alpha)$, then P is recursive and

$$\mathsf{G}(m, \alpha, \lambda p\,.\,\mathsf{H}(p, m, \alpha)) \simeq ((\text{least } w\,.\,\mathsf{P}((w)_0, (w)_1, (w)_2, m, \alpha))_1)_0.$$

Hence if we take F to be the functional defined by the right-hand expression, F has the desired property. □

3.10 Corollary. *The class of recursive functionals is closed under functional substitution. The class of recursive relations and recursive functionals is closed under relational substitution.*

Proof. If H is recursive, then the hypothesis is automatically satisfied. □

3.11–3.13 Exercises

3.11. Show that the class of partial recursive functionals is the smallest class of functionals containing the functionals of clause (0) of Definition 2.1 and closed under functional composition and unbounded search.

3.12. Prove the following effective version of Theorem 3.9: there exists a primitive recursive function f such that for any $a, d, \mathbf{m}$, and $\boldsymbol{\alpha}$ such that $\lambda p\,.\,\{d\}(p, \mathbf{m}, \boldsymbol{\alpha})$ is total,

$$\{f(a, d)\}(\mathbf{m}, \boldsymbol{\alpha}) \simeq \{a\}(\mathbf{m}, \boldsymbol{\alpha}, \lambda p\,.\,\{d\}(p, \mathbf{m}, \boldsymbol{\alpha})).$$

3.13. The definition schema of Exercise 1.14 makes sense if we repace the relation $<$ on ω by any relation $<^*$ such that $\leqslant^*$ is a well-ordering of ω. Show that if $\leqslant^*$ is recursive, then the class of recursive functionals is closed under this schema.

3.14. Notes. The Normal Forms are due to Kleene [1935/36] and are characteristic of Ordinary Recursion Theory as opposed to recursion relative to a functional of type 2 or higher. To appreciate their value, the reader might glance ahead at the work required to prove Corollary VI.2.11, the analogue of Theorem 3.9 (functional substitution) for functionals partial recursive in a type-2 functional.

4. *Semi-Recursive Relations*

We noted following 3.7 that the recursive relations might be viewed as the effectively closed-open relations. In this section we study *effectively open* (semi-recursive) and *effectively closed* (co-semi-recursive) relations.

4.1 Definition. For any R,

(i) R is *semi-recursive* iff $\mathsf{R} = \mathrm{Dm}\,\mathsf{G}$ for some partial recursive functional G; an index for G is called a *semi-index* for R;

(ii) R is *co-semi-recursive* iff $\sim\mathsf{R}$ is semi-recursive.

The semi-recursive relations are, of course, a formal counterpart of the *mechanically semi-calculable* relations. R is semi-calculable iff there is a finite program which directs an idealized computer of the sort discussed in the introduction to this chapter to accept inputs of the form $(\mathbf{m}, \boldsymbol{\alpha})$ and carry out a computation which terminates iff $\mathsf{R}(\mathbf{m}, \boldsymbol{\alpha})$.

4.2 Lemma. *Every recursive relation is semi-recursive but there exist semi-recursive subsets of ω which are not recursive.*

Proof. For any recursive R, define F and G by

$$F(m) = \begin{cases} \text{an index for } \lambda p.0, & \text{if} \quad m = 0; \\ 0, & \text{otherwise}; \end{cases}$$

$$\mathsf{G}(\mathbf{m}, \boldsymbol{\alpha}) \simeq \{F(\mathsf{K}_{\mathsf{R}}(\mathbf{m}, \boldsymbol{\alpha}))\}(0).$$

Then F and G are partial recursive and $\mathsf{R} = \mathrm{Dm}\,\mathsf{G}$.

For the second part we use a diagonal argument. Let $g(a) \simeq \{a\}(a)$ and $A = \mathrm{Dm}\, g$. Clearly g is partial recursive so A is semi-recursive. Suppose, for a contradiction that A were in fact recursive. Then by Theorem 2.9 the function f defined by

$$f(a) = \begin{cases} g(a)+1, & \text{if} \quad a \in A; \\ 0, & \text{otherwise} \end{cases}$$

is recursive. Let b be an index for f. Since $\{b\}$ is total, $b \in A$ and thus

$$f(b) = g(b) + 1 = \{b\}(b) + 1 = f(b) + 1,$$

a contradiction. $\square$

Note the similarity of this proof with that of Theorem 1.11. Essentially the same argument shows that the set of indices of total functions is not a recursive set (Exercise 4.23).

We begin now to study the properties of the class of semi-recursive relations. The first lemma follows directly from the definition.

4.3 Lemma. (i) *The class of semi-recursive relations is closed under finite intersection and bounded universal quantification*;

(ii) *the class of semi-recursive relations and partial recursive functionals is closed under relational composition.*

Proof. If $\mathsf{R} = \mathrm{Dm}\,\mathsf{F}$ and $\mathsf{S} = \mathrm{Dm}\,\mathsf{G}$ with F and G partial recursive, then $\mathsf{R} \cap \mathsf{S}$ is the domain of $\lambda\mathbf{m}\boldsymbol{\alpha}.\,\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}) + \mathsf{G}(\mathbf{m}, \boldsymbol{\alpha})$. If

$$\mathsf{Q}(p, \mathbf{m}, \boldsymbol{\alpha}) \leftrightarrow (\forall q < p)\,\mathsf{S}(q, \mathbf{m}, \boldsymbol{\alpha}),$$

define a partial recursive functional H by

$$\mathsf{H}(0, \mathbf{m}, \boldsymbol{\alpha}) \simeq 0;$$

$$\mathsf{H}(p+1, \mathbf{m}, \boldsymbol{\alpha}) \simeq \mathsf{H}(p, \mathbf{m}, \boldsymbol{\alpha}) + \mathsf{G}(p, \mathbf{m}, \boldsymbol{\alpha}).$$

Then $\mathsf{Q} = \mathrm{Dm}\,\mathsf{H}$ and thus is semi-recursive.

For (ii), if $\mathsf{S} = \mathrm{Dm}\,\mathsf{I}$ and

$$\mathsf{R}(\mathbf{m}, \boldsymbol{\alpha}) \leftrightarrow \mathsf{S}(\mathsf{G}_0(\mathbf{m}, \boldsymbol{\alpha}), \ldots, \mathsf{G}_{k'-1}(\mathbf{m}, \boldsymbol{\alpha}), \boldsymbol{\alpha}),$$

then R is the domain of the partial recursive functional $\mathrm{FCmp}(\mathsf{I}, \mathsf{G})$. $\square$

Note, however, that the *recursive* relations and partial recursive functionals are *not* closed under relational composition (Exercise 4.26).

The remainder of the theory of semi-recursive relations depends on the normal forms of § 3. First, it is an immediate consequence of Theorem 3.9 that the class of semi-recursive relations and recursive functionals is closed under relational substitution.

4.4 Selection Theorem. *There exists a partial recursive functional* Sel *such that for all* $a, \mathbf{m}$, *and* $\boldsymbol{\alpha}$,

(i) $\exists p.\{a\}(p, \mathbf{m}, \boldsymbol{\alpha})\downarrow$ *iff* $\mathrm{Sel}(a, \langle\mathbf{m}\rangle, \langle\boldsymbol{\alpha}\rangle)\downarrow$, *and*

(ii) *if* $\exists p.\{a\}(p, \mathbf{m}, \boldsymbol{\alpha})\downarrow$, *then* $\{a\}(\mathrm{Sel}(a, \langle\mathbf{m}\rangle, \langle\boldsymbol{\alpha}\rangle), \mathbf{m}, \boldsymbol{\alpha})\downarrow$.

Proof. With T as in Theorem 3.1 we take

$$\mathrm{Sel}(a, \langle \mathbf{m}\rangle, \langle \boldsymbol{\alpha}\rangle) \simeq (\text{least } q\,.\,\mathsf{T}(a, \langle (q)_0, \mathbf{m}\rangle, (q)_1, \langle \boldsymbol{\alpha}\rangle))_0. \quad \square$$

4.5 Corollary. *For any semi-recursive relation* R, *there exists a partial recursive functional* $\mathrm{Sel}_{\mathsf{R}}$ *such that for all* $\mathbf{m}$ *and* $\boldsymbol{\alpha}$,

$$\exists p\,.\,\mathsf{R}(p, \mathbf{m}, \boldsymbol{\alpha}) \leftrightarrow \mathsf{R}(\mathrm{Sel}_{\mathsf{R}}(\mathbf{m}, \boldsymbol{\alpha}), \mathbf{m}, \boldsymbol{\alpha}) \leftrightarrow \mathrm{Sel}_{\mathsf{R}}(\mathbf{m}, \boldsymbol{\alpha})\downarrow.$$

Proof. If a is any semi-index for R, take

$$\mathrm{Sel}_{\mathsf{R}}(\mathbf{m}, \boldsymbol{\alpha}) \simeq \mathrm{Sel}(a, \langle \mathbf{m}\rangle, \langle \boldsymbol{\alpha}\rangle).$$

$\mathrm{Sel}_{\mathsf{R}}$ is partial recursive by Theorem 3.9 and clearly has the required property. $\square$

By analogy with the notation "least" we shall sometimes write $\mathrm{Sel}\, p\,.\,\mathsf{R}(p, \mathbf{m}, \boldsymbol{\alpha})$ for $\mathrm{Sel}_{\mathsf{R}}(\mathbf{m}, \boldsymbol{\alpha})$.

4.6 Corollary. *The class of semi-recursive relations of a given rank is closed under finite union.*

Proof. Suppose $\mathsf{R} = \mathrm{Dm}\{a\}$ and $\mathsf{S} = \mathrm{Dm}\{b\}$ are semi-recursive. Let F be the recursive function defined by

$$F(p) = \begin{cases} a, & \text{if } p = 0; \\ b, & \text{if } p > 0. \end{cases}$$

Set

$$\mathsf{H}(p, \mathbf{m}, \boldsymbol{\alpha}) \simeq \{F(p)\}(\mathbf{m}, \boldsymbol{\alpha}).$$

Then $(\mathsf{R} \cup \mathsf{S})(\mathbf{m}, \boldsymbol{\alpha}) \leftrightarrow \exists p\,.\,\mathsf{H}(p, \mathbf{m}, \boldsymbol{\alpha})\downarrow$ so that $\mathsf{R} \cup \mathsf{S} = \mathrm{Dm}(\mathrm{Sel}_{\mathrm{Dm}\,\mathsf{H}})$ and is thus semi-recursive. $\square$

4.7 Definition ((*unbounded*) *number quantification*). If R is any relation of rank $(k+1, l)$ then $\exists^0\mathsf{R}$ and $\forall^0\mathsf{R}$ are the relations P and Q of rank (k, l) such that

$$\mathsf{P}(\mathbf{m}, \boldsymbol{\alpha}) \leftrightarrow \exists p\,.\,\mathsf{R}(p, \mathbf{m}, \boldsymbol{\alpha}),$$

and

$$\mathsf{Q}(\mathbf{m}, \boldsymbol{\alpha}) \leftrightarrow \forall p\,.\,\mathsf{R}(p, \mathbf{m}, \boldsymbol{\alpha}).$$

4.8 Corollary. *The class of semi-recursive relations is closed under existential number quantification* ($\exists^0$), *hence also under bounded existential quantification* ($\exists^0_<$).

Proof. Immediate from Corollary 4.5. $\square$

It follows, of course, that the co-semi-recursive relations are closed under $\forall^0$ and $\forall^0_<$. The semi-recursive relations are *not*, however, closed under $\forall^0$ (Exercise 4.24).

To extend definition by cases to semi-recursive relations we must modify the definition (cf. Exercise 4.28). We say that F is *defined by positive cases* from $\mathsf{G}_0, \ldots, \mathsf{G}_{k'-1}$ and $\mathsf{R}_0, \ldots, \mathsf{R}_{k'-1}$ if all are of rank (k, l), for any $(\mathbf{m}, \boldsymbol{\alpha}) \in {}^{k,l}\omega$ there is at most one $i < k'$ such that $\mathsf{R}_i(\mathbf{m}, \boldsymbol{\alpha})$, and

$$\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}) \simeq \begin{cases} \mathsf{G}_0(\mathbf{m}, \boldsymbol{\alpha}), & \text{if} \quad \mathsf{R}_0(\mathbf{m}, \boldsymbol{\alpha}); \\ \quad \vdots & \\ \mathsf{G}_{k'-1}(\mathbf{m}, \boldsymbol{\alpha}), & \text{if} \quad \mathsf{R}_{k'-1}(\mathbf{m}, \boldsymbol{\alpha}); \\ \text{undefined, otherwise.} & \end{cases}$$

4.9 Theorem. *The class of partial recursive functionals and semi-recursive relations is closed under definition by positive cases.*

Proof. Let f and g be recursive functions defined by

$$f(i) = \begin{cases} \text{a semi-index for } \mathsf{R}_i, & \text{if} \quad i < k'; \\ 0, \quad \text{otherwise;} & \end{cases}$$

$$g(i) = \begin{cases} \text{an index for } \mathsf{G}_i, & \text{if} \quad i < k'; \\ 0, \quad \text{otherwise.} & \end{cases}$$

Then if

$$\mathsf{H}(i, \mathbf{m}, \boldsymbol{\alpha}) \simeq \{f(i)\}(\mathbf{m}, \boldsymbol{\alpha}),$$

it is easy to check that the desired F is given by

$$\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}) \simeq \{g(\mathrm{Sel}_{\mathrm{Dm}\,\mathsf{H}}(\mathbf{m}, \boldsymbol{\alpha}))\}(\mathbf{m}, \boldsymbol{\alpha}). \quad \square$$

4.10 Corollary. *A relation is recursive iff it is both semi-recursive and co-semi-recursive.*

Proof. If R is recursive so is $\sim\mathsf{R}$ and thus R is both semi-recursive and co-semi-recursive by Lemma 4.2. Conversely, if both R and $\sim\mathsf{R}$ are semi-recursive, then

$$\mathsf{K}_\mathsf{R}(\mathbf{m}, \boldsymbol{\alpha}) \simeq \begin{cases} 0, & \text{if} \quad \mathsf{R}(\mathbf{m}, \boldsymbol{\alpha}); \\ 1, & \text{if} \quad \sim\mathsf{R}(\mathbf{m}, \boldsymbol{\alpha}); \end{cases}$$

is a definition by positive semi-recursive cases and thus K_R is recursive. □

4.11 Corollary. *For any partial functional* F, F *is partial recursive iff its graph* Gr_F *is semi-recursive, and* F *is recursive iff* F *is total and* Gr_F *is recursive.*

Proof. If F is partial recursive, then since $=$ is recursive and

$$\mathrm{Gr}_\mathsf{F}(n, \mathbf{m}, \boldsymbol{\alpha}) \leftrightarrow \mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}) = n,$$

Gr_F is semi-recursive by 4.3(ii). For the converse we have

$$\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}) \simeq \mathrm{Sel}_{\mathrm{Gr}(\mathsf{F})}(\mathbf{m}, \boldsymbol{\alpha}).$$

The second part follows from the observation that if F is total, $\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}) \not\simeq n$ iff $(\exists n' \neq n)\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}) \simeq n'$. □

To this point we have used the normal forms only to prove the Selection Theorem. Now we use them to obtain some new characterizations of the class of semi-recursive relations.

4.12 Theorem. *For any relation* R, *the following are equivalent*:

(i) R *is semi-recursive*;

(ii) *for some (primitive) recursive relation* S, $\mathsf{R}(\mathbf{m}, \boldsymbol{\alpha}) \leftrightarrow \exists p\, \mathsf{S}(p, \mathbf{m}, \boldsymbol{\alpha})$;

(iii) *for some (primitive) recursive relation* S,

$$\mathsf{R}(\mathbf{m}, \boldsymbol{\alpha}) \leftrightarrow \exists p\, S(\mathbf{m}, \bar{\alpha}_0(p), \ldots, \bar{\alpha}_{l-1}(p)) \text{ (only when } l > 0).$$

Proof. Suppose R is semi-recursive, say with semi-index a. Then

$$\mathsf{R}(\mathbf{m}, \boldsymbol{\alpha}) \leftrightarrow \exists n . \{a\}(\mathbf{m}, \boldsymbol{\alpha}) \simeq n \leftrightarrow \exists u\, \mathsf{T}(a, \langle \mathbf{m} \rangle, u, \langle \boldsymbol{\alpha} \rangle)$$
$$\leftrightarrow \exists u\, T(a, \langle \mathbf{m} \rangle, u, \langle \bar{\alpha}_0(u), \ldots, \bar{\alpha}_{l-1}(u) \rangle)$$

by Theorems 3.1 and 3.5. Hence (ii) and (iii) follow if we set

$$\mathsf{S}(p, \mathbf{m}, \boldsymbol{\alpha}) \leftrightarrow \mathsf{T}(a, \langle \mathbf{m} \rangle, p, \langle \boldsymbol{\alpha} \rangle)$$

and

$$S(\mathbf{m}, s_0, \ldots, s_{l-1}) \leftrightarrow T(a, \langle \mathbf{m} \rangle, \lg(s_0), \langle s_0, \ldots, s_{l-1} \rangle).$$

On the other hand, if (ii) holds (in either form) then R is the domain of the function $\lambda \mathbf{m}\boldsymbol{\alpha}.\,\mathrm{least}\,p\,.\,\mathsf{S}(p, \mathbf{m}, \boldsymbol{\alpha})$ and is thus semi-recursive. The implication (iii)$\rightarrow$(i) is similar. $\square$

Of course, combinations of forms (ii) and (iii) are possible. For example, every semi-recursive R may also be expressed in the form

$$\mathsf{R}(\mathbf{m}, \boldsymbol{\alpha}) \leftrightarrow \exists p\ \mathsf{S}(\bar{\alpha}_0(p), \mathbf{m}, \bar{\alpha}_1(p), \alpha_2, \ldots, \alpha_{l-1}).$$

4.13 Definition (*function quantification*). For any relation R of rank $(k, l+1)$, $\exists^1\mathsf{R}$ and $\forall^1\mathsf{R}$ are the relations P and Q of rank (k, l) such that

$$\mathsf{P}(\mathbf{m}, \boldsymbol{\alpha}) \leftrightarrow \exists \beta\ \mathsf{R}(\mathbf{m}, \boldsymbol{\alpha}, \beta),$$

and

$$\mathsf{Q}(\mathbf{m}, \boldsymbol{\alpha}) \leftrightarrow \forall \beta\ \mathsf{R}(\mathbf{m}, \boldsymbol{\alpha}, \beta).$$

4.14 Theorem. *The class of semi-recursive relations is closed under existential function quantification* ($\exists^1$).

Proof. Suppose R is semi-recursive. Then by 4.12 there exists a recursive S such that $\mathsf{R}(\mathbf{m}, \boldsymbol{\alpha}, \beta) \leftrightarrow \exists p\,.\,\mathsf{S}(\bar{\beta}(p), \mathbf{m}, \boldsymbol{\alpha})$. Then

$$\begin{aligned} \exists \beta\ \mathsf{R}(\mathbf{m}, \boldsymbol{\alpha}, \beta) &\leftrightarrow \exists \beta\ \exists p\ \mathsf{S}(\bar{\beta}(p), \mathbf{m}, \boldsymbol{\alpha}) \\ &\leftrightarrow \exists s\ \mathsf{S}(s, \mathbf{m}, \boldsymbol{\alpha}). \end{aligned}$$

Thus $\exists^1\mathsf{R} = \exists^0\mathsf{S}$ and is thus semi-recursive by 4.12. $\square$

4.15. Theorem. *For any non-empty set $A \subseteq \omega$, the following are equivalent*:

(i) *A is semi-recursive*;
(ii) $A = \mathrm{Im}\,F$, *for some recursive function F*;
(iii) $A = \mathrm{Im}\,\mathsf{F}$, *for some recursive functional* F;
(iv) $A = \mathrm{Im}\,\mathsf{F}$, *for some partial recursive functional* F.

Proof. Suppose first that A is semi-recursive, so by Theorem 4.12 there exists a recursive relation S such that

$$n \in A \leftrightarrow \exists p\,.\,S(p, n).$$

Then if $\bar{n}$ is some fixed element of A, A is the image of the recursive function F defined by

$$F(m) = \begin{cases} (m)_1, & \text{if} \quad S((m)_0, (m)_1); \\ \bar{n}, & \text{otherwise.} \end{cases}$$

The implications (ii)$\rightarrow$(iii)$\rightarrow$(iv) are trivial. Suppose $A = \operatorname{Im} \mathsf{F}$ for a partial recursive F. Then

$$n \in A \leftrightarrow \exists \mathbf{m} \exists \boldsymbol{\alpha} . \operatorname{Gr}_{\mathsf{F}}(n, \mathbf{m}, \boldsymbol{\alpha}).$$

$\operatorname{Gr}_{\mathsf{F}}$ is semi-recursive by Corollary 4.11 and thus so is A by 4.8 and 4.14. □

Because of this theorem, semi-recursive subsets of ω are often called *recursively enumerable.*

We conclude this section with a discussion of two notions which originated in descriptive set theory and have played a significant role in recursion theory as well.

4.16 Definition. A class X of relations has the *reduction property* iff for any R and S belonging to X, there exist R^* and S^* in X such that:

(i) $\mathsf{R}^* \subseteq \mathsf{R}$ and $\mathsf{S}^* \subseteq \mathsf{S}$;

(ii) $\mathsf{R}^* \cap \mathsf{S}^* = \varnothing$;

(iii) $\mathsf{R}^* \cup \mathsf{S}^* = \mathsf{R} \cup \mathsf{S}$.

The pair $(\mathsf{R}^*, \mathsf{S}^*)$ is said to *reduce* (R, S).

4.17 Theorem. *The class of semi-recursive relations has the reduction property.*

Proof. Let R and S be any two semi-recursive relations. If R and S are not of the same rank, then they are disjoint and it suffices to take $\mathsf{R}^* = \mathsf{R}$ and $\mathsf{S}^* = \mathsf{S}$. Otherwise by Theorem 4.12 there exist recursive relations P and Q such that $\mathsf{R} = \exists^0 \mathsf{P}$ and $\mathsf{S} = \exists^0 \mathsf{Q}$. We set

$$\mathsf{R}^*(\mathbf{m}, \boldsymbol{\alpha}) \leftrightarrow \exists p\, [\mathsf{P}(p, \mathbf{m}, \boldsymbol{\alpha}) \wedge (\forall q < p) \sim \mathsf{Q}(q, \mathbf{m}, \boldsymbol{\alpha})];$$

$$\mathsf{S}^*(\mathbf{m}, \boldsymbol{\alpha}) \leftrightarrow \exists q\, [\mathsf{Q}(q, \mathbf{m}, \boldsymbol{\alpha}) \wedge (\forall p \leqslant q) \sim \mathsf{P}(p, \mathbf{m}, \boldsymbol{\alpha})].$$

Again by Theorem 4.12, R^* and S^* are semi-recursive, and it is straightforward to check that conditions (i)–(iii) are satisfied. □

4.18 Definition. A class X of relations has the *separation property* iff for all R and S belonging to X, if R and S are of the same rank and $\mathsf{R} \cap \mathsf{S} = \varnothing$, then there exists a relation P such that both P and $\sim\mathsf{P}$ belong to X and $\mathsf{R} \subseteq \mathsf{P} \subseteq \sim\mathsf{S}$. P is said to *separate* R and S.

These two properties are closely related. First we have

4.19 Lemma. *For any class X of relations, if X has the reduction property, then $cX = \{\sim\mathsf{R}\colon \mathsf{R} \in X\}$ has the separation property.*

Proof. Suppose X has the reduction property, R and S belong to X and have the same rank (k, l), and $(\sim\mathsf{R}) \cap (\sim\mathsf{S}) = \varnothing$. Then $\mathsf{R} \cup \mathsf{S} = {}^{k,l}\omega$. Let $(\mathsf{R}^*, \mathsf{S}^*)$ reduce (R, S). Since also $\mathsf{R}^* \cup \mathsf{S}^* = {}^{k,l}\omega$ and $\mathsf{R}^* \cap \mathsf{S}^* = \varnothing$, $\mathsf{R}^* = \sim\mathsf{S}^*$ so both S^* and $\sim\mathsf{S}^*$ belong to X. Also, $\sim\mathsf{R} \subseteq \sim\mathsf{R}^* = \mathsf{S}^* \subseteq \mathsf{S} = \sim(\sim\mathsf{S})$, so S^* separates $\sim\mathsf{R}$ and $\sim\mathsf{S}$. □

4.20 Definition. A class X of relations is *indexable* iff there exists a relation $\mathsf{U} \in X$ such that for every $\mathsf{R} \in X$, there exists a number a such that

$$\mathsf{R}(\mathbf{m}, \boldsymbol{\alpha}) \leftrightarrow \mathsf{U}(a, \langle \mathbf{m} \rangle, \langle \boldsymbol{\alpha} \rangle).$$

U is said to be *universal* for X.

4.21 Lemma. *For any indexable class X of relations which is closed under composition with recursive functionals, if X has the reduction property, then X does not have the separation property.*

Proof. Let U be universal for X and set

$$R(m) \leftrightarrow \mathsf{U}((m)_0, \langle m \rangle, \langle\ \rangle) \quad \text{and} \quad S(m) \leftrightarrow \mathsf{U}((m)_1, \langle m \rangle, \langle\ \rangle).$$

R and S both belong to X so there exists (R^*, S^*) which reduces (R, S). Suppose that X has the separation property and that P separates R^* and S^*. Since both P and $\sim P \in X$, there exist a and b such that

$$P(m) \leftrightarrow \mathsf{U}(a, \langle m \rangle, \langle\ \rangle) \quad \text{and} \quad \sim P(m) \leftrightarrow \mathsf{U}(b, \langle m \rangle, \langle\ \rangle).$$

Consider the number $m = \langle b, a \rangle$. We have

(1) $$P(\langle b, a \rangle) \leftrightarrow \mathsf{U}(a, \langle\langle b, a \rangle\rangle, \langle\ \rangle) \leftrightarrow S(\langle b, a \rangle),$$

and

(2) $$\sim P(\langle b, a \rangle) \leftrightarrow \mathsf{U}(b, \langle\langle b, a \rangle\rangle, \langle\ \rangle) \leftrightarrow R(\langle b, a \rangle).$$

Suppose that $P(\langle b, a \rangle)$. Since $P \cap S^* = \varnothing$, it follows from (1) that $\langle b, a \rangle \in S \sim S^*$. Then $R^*(\langle b, a \rangle)$ so $R(\langle b, a \rangle)$ and hence by (2), $\sim P(\langle b, a \rangle)$, a contradiction. A similar argument obtains a contradiction from the hypothesis $\sim P(\langle b, a \rangle)$. □

4.22 Corollary. (i) *The class of semi-recursive relations has the reduction property but not the separation property*;

(ii) *the class of co-semi-recursive relations has the separation property but not the reduction property.*

Proof. The class of semi-recursive relations has the universal relation

$$\mathsf{U}(a, \langle \mathbf{m} \rangle, \langle \boldsymbol{\alpha} \rangle) \leftrightarrow \{a\}(\mathbf{m}, \boldsymbol{\alpha})\downarrow. \quad \square$$

Note that in view of Corollary 4.10, the separation property for the class of (co-) semi-recursive relations asserts separability by a *recursive* relation. The second part of 4.22(i) is often referred to as establishing the existence of a pair of *recursively inseparable* semi-recursive relations.

4.23–4.34 Exercises

4.23. Show that the set of indices a such that $\{a\}$ is a total unary function is not recursive or even semi-recursive.

4.24. Show that the class of semi-recursive relations is not closed under $\forall^0$.

4.25. Let $\mathrm{Sel}'_{\mathsf{R}}(\mathbf{m}, \boldsymbol{\alpha}) \simeq \text{least } p \,.\, \mathsf{R}(p, \mathbf{m}, \boldsymbol{\alpha})$. Show that for some semi-recursive relations R, $\mathrm{Sel}'_{\mathsf{R}}$ is not partial recursive.

4.26. Show that the class of recursive relations and partial recursive functionals is *not* closed under relational composition.

4.27. Show that the class of partial recursive functionals is *not* closed under functional substitution (cf. Theorem 3.9).

4.28. Show that the class of partial recursive functionals and semi-recursive relations is *not* closed under definition by (not necessarily positive) cases.

4.29. Prove or refute the following analogue of Corollary 4.11: for any total functional F, F is primitive recursive iff Gr_{F} is primitive recursive.

4.30. Show that a non-empty subset A of ω is semi-recursive iff A is the image of a primitive recursive function f. Show that an infinite subset A of ω is semi-recursive iff A is the image of a one-one recursive function F.

4.31. For any partial function f of rank 1, let $\Omega[f]$ be defined as is Ω with the following additional clause:

$$\text{if} \quad f(p) \simeq n, \quad \text{then} \quad (\langle 0, k+1, l, 6 \rangle, p, \mathbf{m}, \boldsymbol{\alpha}, n) \in \Omega[f].$$

We write

$$\{a\}(\mathbf{m}, \boldsymbol{\alpha}, f) \simeq n \leftrightarrow (a, \mathbf{m}, \boldsymbol{\alpha}, n) \in \Omega[f].$$

We have thus defined the class of partial recursive functionals with arguments of type $(\mathbf{m}, \boldsymbol{\alpha}, f)$. Show that for any such partial recursive functional F

(i) there exists an ordinary partial recursive functional G such that

$$\mathsf{G}(e, \mathbf{m}, \boldsymbol{\alpha}) \simeq \mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}, \{e\});$$

(ii) (*First Recursion Theorem*) there exists a partial recursive function $\bar{f}$ such that for all p, $\mathsf{F}(p, \bar{f}) \simeq \bar{f}(p)$ and for any h, if also for all p, $\mathsf{F}(p, h) \simeq h(p)$, then $\bar{f} \subseteq h$.

(iii) Formulate and prove a version of (ii) which allows for the presence of parameters.

Hint. For (i) use an analogue of the Second Normal Form of § 3. For (ii), let $g_0 = \varnothing$ and $g_{r+1}(p) \simeq \mathsf{F}(p, g_r)$. Take $\bar{f}(p) \simeq n \leftrightarrow \exists r \,.\, g_r(p) \simeq n$.

4.32. Use the techniques of Exercise I.2.7 to show that the class of closed subsets of ${}^{\omega}\omega$ has the separation property. (Show that for any A and B, if $\forall n \,.\, \mathsf{A}^{((n))}$ and $\mathsf{B}^{((n))}$ can be separated by a Kalmar set, then also A and B can be separated by a Kalmar set.)

4.33. Give an alternate proof using the Selection Theorem that the class of semi-recursive relations has the reduction property.

4.34. A class of relations X is said to have the *second separation property* iff for any $\mathsf{R}, \mathsf{S} \in X$ of the same rank, there exist R^* and S^* such that $\sim\mathsf{R}^*$ and $\sim\mathsf{S}^* \in X$ and $\mathsf{R} \sim \mathsf{S} \subseteq \mathsf{R}^* \subseteq \sim\mathsf{S}^* \subseteq \sim(\mathsf{S} \sim \mathsf{R})$. Show that if cX has the reduction property, then X has the second separation property.

4.35 Notes. The remarks made in the Notes to the preceding section apply also to the Selection Theorem and its consequences — all of § VI.3 is devoted to a technical lemma needed to prove the Selection Theorem for functionals partial recursive in a type-2 functional (VI.4.1).

Theorem 4.12 does not depend on any of the earlier results of this section and could be used to give alternate proofs for many of them. For example, if R and R' are semi-recursive and S and S' are recursive relations such that $\mathsf{R} = \exists^0\mathsf{S}$ and $\mathsf{R}' = \exists^0\mathsf{S}'$, then $\mathsf{R} \cup \mathsf{R}' = \exists^0(\mathsf{S} \cup \mathsf{S}')$ and is thus also semi-recursive. Similarly, if $\mathsf{R}(q, \mathbf{m}, \boldsymbol{\alpha}) \leftrightarrow \exists p\, \mathsf{S}(p, q, \mathbf{m}, \boldsymbol{\alpha})$, $\exists q\, \mathsf{R}(q, \mathbf{m}, \boldsymbol{\alpha}) \leftrightarrow \exists r\, \mathsf{S}((r)_0, (r)_1, \mathbf{m}, \boldsymbol{\alpha})$, so that if R is semi-recursive, so is $\exists^0\mathsf{R}$. We have given the proofs as we have in order to emphasize the parallel with the generalized recursion theories of Chapters VI and VII where the analogues of 4.2–4.11 hold, but that of 4.12 does not.

The term "semi-recursive" is relatively new and many authors still use "recursively enumerable" (r.e.) even for subsets of ${}^{k,l}\omega$ (which are not enumerable at all!). The need for separate terms, even for sets of numbers, becomes especially clear in § VI.1 where the class of "recursively-in-E enumerable" sets of numbers (images of functions partial recursive in E) is properly included in the class of sets semi-recursive in E (domains of functions partial recursive in E). To complete the confusion, at least one author has used "semi-recursive" in a totally different sense from ours.

5. Relativization

In our description of mechanical calculability, the input to the idealized computer consisted of an infinite memory device which contained the arguments and could be connected to the computer. In this section we consider the notion of calculability which arises from considering a second such memory device with fixed content permanently connected to the computer. If this device contains the graphs of a sequence of functions $\boldsymbol{\beta}$, then we say that a functional computed by the machine is *calculable in* (or *relative to*) $\boldsymbol{\beta}$.

5.1 Definition. For any F, R, and $\boldsymbol{\beta}$,

(i) F is *partial recursive in* $\boldsymbol{\beta}$ iff there exists a partial recursive G such that $\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}) \simeq \mathsf{G}(\mathbf{m}, \boldsymbol{\alpha}, \boldsymbol{\beta})$; an index for G is also called an *index for* F *from* $\boldsymbol{\beta}$;

(ii) F is *recursive in* $\boldsymbol{\beta}$ iff F is partial recursive in $\boldsymbol{\beta}$ and total;

(iii) R is *recursive in* $\boldsymbol{\beta}$ iff K_R is recursive in $\boldsymbol{\beta}$; an index for K_R from $\boldsymbol{\beta}$ is also called an *index for* R *from* $\boldsymbol{\beta}$;

(iv) R is *semi-recursive in* $\boldsymbol{\beta}$ iff R is the domain of some functional G partial recursive in $\boldsymbol{\beta}$; an index for G from $\boldsymbol{\beta}$ is also called a *semi-index for* R *from* $\boldsymbol{\beta}$.

In accord with the conventions of § I.1 we also say that F is partial recursive in $(A_0, \ldots, A_r)$ iff F if partial recursive in $(\mathsf{K}_{A_0}, \ldots, \mathsf{K}_{A_r})$, F is partial recursive in R iff F is partial recursive in $\{\langle\mathbf{m}\rangle: R(\mathbf{m})\}$, etc.

Note that we did *not* define F to be recursive in $\boldsymbol{\beta}$ just in case $\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}) = \mathsf{G}(\mathbf{m}, \boldsymbol{\alpha}, \boldsymbol{\beta})$ for some *recursive* (total) G. This is a more restrictive notion (cf. Exercises 5.8–9). From (iv) it is clear that R is semi-recursive in $\boldsymbol{\beta}$ iff $\mathsf{R}(\mathbf{m}, \boldsymbol{\alpha}) \leftrightarrow \mathsf{S}(\mathbf{m}, \boldsymbol{\alpha}, \boldsymbol{\beta})$ for some semi-recursive S, but again it is false that R is recursive in $\boldsymbol{\beta}$ iff this equivalence holds for a recursive S.

If we replace "primitive recursive" by "primitive recursive in $\boldsymbol{\beta}$" (cf. Exercise 1.18), "partial recursive" by "partial recursive in $\boldsymbol{\beta}$", etc., the theory of §§ 1–4 all goes through with essentially the same proofs. We shall refer to these results as the *relativized versions* of the various theorems.

5.2 Lemma (Transitivity). *If all of* $\beta_0, \ldots, \beta_r$ *are recursive* (*in* γ), *and* F *is partial recursive in* $\boldsymbol{\beta}$, *then* F *is partial recursive* (*in* γ).

Proof. We prove the version with γ. By assumption there exist partial recursive $\mathsf{G}, \mathsf{H}_0, \ldots, \mathsf{H}_r$ such that

$$\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}) \simeq \mathsf{G}(\mathbf{m}, \boldsymbol{\alpha}, \boldsymbol{\beta}), \quad \text{and for all} \quad i \leq r, \quad \beta_i(p) = \mathsf{H}_i(p, \boldsymbol{\gamma}).$$

By Theorem 3.9 there exists a partial recursive functional I such that for all $\boldsymbol{\gamma}$ such that $\lambda p \,.\, \mathsf{H}_i(p, \boldsymbol{\gamma})$ is total for all $i \leq r$,

$$\mathsf{I}(\mathbf{m}, \boldsymbol{\alpha}, \boldsymbol{\gamma}) \simeq \mathsf{G}(\mathbf{m}, \boldsymbol{\alpha}, \lambda p \,.\, \mathsf{H}_0(p, \boldsymbol{\gamma}), \ldots, \lambda p \,.\, \mathsf{H}_r(p, \boldsymbol{\gamma})).$$

In particular this is true for the fixed $\boldsymbol{\gamma}$ under consideration and thus $\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}) \simeq \mathsf{I}(\mathbf{m}, \boldsymbol{\alpha}, \boldsymbol{\gamma})$. Hence F is partial recursive in $\boldsymbol{\gamma}$. □

It follows that the relation "recursive in" is transitive on members of ${}^{\omega}\omega$. For any α, the *degree* of α is defined by

$$\mathrm{dg}(\alpha) = \{\beta\colon \alpha \text{ is recursive in } \beta \text{ and } \beta \text{ is recursive in } \alpha\}.$$

The degrees inherit a partial ordering — we say

$$\mathrm{dg}(\alpha) \leq \mathrm{dg}(\beta) \leftrightarrow \alpha \text{ is recursive in } \beta;$$

$$\mathrm{dg}(\alpha) < \mathrm{dg}(\beta) \leftrightarrow \mathrm{dg}(\alpha) \leq \mathrm{dg}(\beta) \quad \text{and} \quad \mathrm{dg}(\beta) \not\leq \mathrm{dg}(\alpha).$$

Because of transitivity these notions do not depend on the choice of representative. This is far from being a linear ordering. Because there are only countably many partial recursive functionals, $\{\beta\colon \beta \text{ is recursive in } \alpha\}$ is always countable.

5.3 Theorem. *For any non-recursive α, $\{\beta\colon \alpha$ is recursive in $\beta\}$ is meager.*

Proof. Let α be a fixed non-recursive function, $\mathsf{A} = \{\beta\colon \alpha \text{ is recursive in } \beta\}$, and for each a, $\mathsf{A}_a = \{\beta\colon \alpha \text{ is recursive in } \beta \text{ with index } a\}$. If A is not meager, then for some a and some p, A_a is dense in $[p]$. We claim, however, that from this follows that

$$\alpha(m) = n \leftrightarrow \exists u \, \exists s \, [(p \subseteq s \vee s \subseteq p) \wedge \mathrm{lg}(s) = u \wedge T(a, \langle m \rangle, u, \langle s \rangle) \wedge (u)_0 = n].$$

This implies, by Corollary 4.11, that α is recursive, contrary to assumption.

Suppose first that $\alpha(m) = n$. Since A_a is dense in $[p]$ there exists a $\beta \in [p]$ such that $\{a\}(m, \beta) \simeq n$. Then $\exists u \, [T(a, \langle m \rangle, u, \langle \bar{\beta}(u) \rangle) \wedge (u)_0 = n]$ and clearly $\bar{\beta}(u) \subseteq p$ or $p \subseteq \bar{\beta}(u)$, and $\mathrm{lg}(\bar{\beta}(u)) = u$.

Conversely suppose there are u and s such that $(p \subseteq s \vee s \subseteq p)$, $\mathrm{lg}(s) = u$, $T(a, \langle m \rangle, u, \langle s \rangle)$, and $(u)_0 = n$. Let β be any element of A_a and the smaller of $[p]$

and $[\bar{\beta}(u)]$ so that $s = \bar{\beta}(u)$. Thus $T(a, \langle m \rangle, u, \langle \bar{\beta}(u) \rangle)$ and $(u)_0 = n$ so $\{a\}(m, \beta) \simeq n$ and hence also $\alpha(m) = n$. □

5.4 Corollary. *For any non-recursive α, there are uncountably many β such that neither α is recursive in β nor β is recursive in α.*

Proof. By 5.3 and the Baire Category Theorem (I.2.2). □

One of the main themes of this book is the notion of complexity for functionals and relations. The simplest objects beyond the finite that we have discussed are the primitive recursive ones. Recursive functionals and relations are more complex in that an additional technique of calculation (the self-determination principle or unbounded search) is needed to compute them. Partial recursive functionals and semi-recursive relations are still more complex in that they are not completely calculable — if $(\mathbf{m}, \boldsymbol{\alpha}) \in \mathrm{Dm}\,\mathsf{F}$, that fact will become known at the conclusion of the computation of $\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha})$, but if $(\mathbf{m}, \boldsymbol{\alpha}) \notin \mathrm{Dm}\,\mathsf{F}$, the attempt to compute $\mathsf{F}(\mathbf{m}, \boldsymbol{\alpha})$ will go on forever (cf. Lemma 4.2).

The relationship "recursive in" is a measure of *relative* complexity among elements of ${}^{\omega}\omega$ and $\mathbf{P}(\omega)$. If α is recursive in β, then "given" the graph of β, α is mechanically calculable. Thus in some sense β is at least as complex as α and more complex if also β is not recursive in α. According to this measure, the recursive functions are the simplest ones, since any recursive function is recursive in every β. Of course, by the preceding corollary, the complexity of two functions may be incomparable.

For any A and $B \subseteq \omega$, if A is recursive and B is semi-recursive but not recursive, then clearly A is recursive in B but B is not recursive in A. Thus in this sense also semi-recursive sets are more complex than recursive sets. It is natural to ask if all non-recursive semi-recursive sets are equally complex. This is answered by:

Friedberg-Mučnik Theorem. There exist semi-recursive sets A and B such that neither A is recursive in B nor B is recursive in A. This result belongs to the theory of degrees, which will be treated in a separate volume in this series Lerman [198?], and we shall not give the proof.

We mentioned in the Introduction the fact that many of the ideas of hierarchy theory arose independently in Descriptive Set Theory and Recursion Theory, and that the fruitful methods of investigation in the two fields are very similar. The following theorem, which characterizes the topology of ${}^{k,l}\omega$ in recursion-theoretic terms, is the basis for this similarity.

5.5 Theorem. *For any* F *and* R,

(i) F *is partial continuous iff* F *is partial recursive in some* β;

(ii) R *is open iff* R *is semi-recursive in some* β;

(iii) R *is closed iff* R *co-semi-recursive in some* β;
(iv) R *is closed-open iff* R *is recursive in some* β.

Proof. Clauses (ii)–(iv) are immediate from (i). The implication ($\leftarrow$) of (i) is the relativized version of Corollary 3.7. Suppose that F is partial continuous; for simplicity take F to be of rank (1,1). Define β by

$$\beta(\langle m, p\rangle) = \begin{cases} n+1, & \text{if } \mathsf{F}(m,\alpha) \simeq n \text{ for all } \alpha \in [p]; \\ 0, & \text{if there is no such } n. \end{cases}$$

The continuity of F implies that for all m and α,

$$\mathsf{F}(m,\alpha)\downarrow \leftrightarrow \exists k\,.\,\beta(\langle m, \bar{\alpha}(k)\rangle) > 0.$$

Hence

$$\mathsf{F}(m,\alpha) \simeq \beta(\langle m, \bar{\alpha}(\text{least } k\,.\,\beta(\langle m, \bar{\alpha}(k)\rangle) > 0)\rangle) \dot{-} 1$$

and is thus partial recursive in β. $\square$

The relationship "recursive in" is by no means the only one which compares the complexity of two objects. The next chapter will provide many examples, but one is appropriate here. A relation R is called (*many-one*) *reducible* to a set A ($\mathsf{R} \leqslant A$) iff for some recursive functional F,

$$\mathsf{R}(\mathbf{m}, \boldsymbol{\alpha}) \leftrightarrow \mathsf{F}(\mathbf{m}, \boldsymbol{\alpha}) \in A.$$

From the composability of partial recursive functionals it follows that if R is many-one reducible to A, then R is recursive in A and if A is (semi-) recursive, so is R (cf. Exercise 5.13).

Among the sets semi-recursive in a given set A there is one which under comparison by many-one reducibility is most complex:

5.6 Definition. For any $A \subseteq \omega$, the (*ordinary*) *jump* of A is the set

$$A^{\mathrm{oJ}} = \{\langle a, \mathbf{m}\rangle : \{a\}(\mathbf{m}, A)\downarrow\}.$$

Clearly A^{oJ} is semi-recursive in A but not recursive in A — in fact,

5.7 Theorem. *For any* $A \subseteq \omega$, *any* k, *and any* $R \subseteq {}^k\omega$,

$$R \text{ is semi-recursive in } A \leftrightarrow R \leqslant A^{\mathrm{oJ}}.$$

Proof. Suppose that R is semi-recursive in A with semi-index a from A. Then

$$R(\mathbf{m}) \leftrightarrow \{a\}(\mathbf{m}, A)\downarrow \leftrightarrow \langle a, \mathbf{m}\rangle \in A^{\text{oJ}}$$

and thus $R \leqslant A^{\text{oJ}}$. On the other hand, A^{oJ} is semi-recursive in A by its definition and hence so is any R many-one reducible to it. □

The jump may also be applied to functions:

$$\beta^{\text{oJ}}(\langle a, \mathbf{m}\rangle) = \begin{cases} 0, & \text{if } \{a\}(\mathbf{m}, \beta)\downarrow; \\ 1, & \text{otherwise;} \end{cases}$$

and $\beta^{\text{oJ}}(p) = 0$ if p is not of the form $\langle a, \mathbf{m}\rangle$. Corresponding to Theorem 5.7 we have only the implication:

$$R \text{ is semi-recursive in } \beta \rightarrow R \text{ is recursive in } \beta^{\text{oJ}}.$$

In each of these results the restriction to relations on numbers is a necessary one. Consider the relation (set) $\mathsf{R} = \{\alpha : \exists m \,.\, \alpha(m) = 0\}$. R is easily seen to be semi-recursive (and hence open) but not closed. Then by Theorem 5.5, R is not recursive in *any* β.

This restriction is closely related to the fact that we have defined here only the notion "F is partial recursive in β" rather than the more general "F is partial recursive in I". This latter notion is the subject of Chapter VI, and we shall have there a version of Theorem 5.7 for arbitrary relations (VI.1.11).

5.8–5.14 Exercises

5.8. A is called *truth-table reducible* to B iff there exist recursive F and D such that

$$m \in A \leftrightarrow \bar{\mathsf{K}}_B(F(m)) \in D.$$

Show that A is truth-table reducible to B iff there exists a (total) recursive functional G such that $\mathsf{K}_A(m) = \mathsf{G}(m, \mathsf{K}_B)$. (Suppose such a G exists and R is a recursive relation such that

$$\mathsf{G}(m, \alpha) = q \leftrightarrow \exists p \,.\, R(\bar{\alpha}(p), m, q).$$

If α^* ranges over ${}^{\omega}2$, we have

$$\forall m\, \forall \alpha^*\, \exists p\, \exists q\, R(\bar{\alpha}^*(p), m, q).$$

Apply the result of Exercise I.2.6.)

5.9. Show that there exist A and B such that A is recursive in B but A is not truth-table reducible to B. (Let

$$a \in A \leftrightarrow \{a\}(a) \quad \text{is defined,}$$

choose by Exercise 4.30 a one-one recursive function f which enumerates A and set

$$n \in B \leftrightarrow (\exists p > n) \,.\, f(p) < f(n).$$

Show first that $\sim B$ is infinite and

$$\text{if} \quad n \notin B \quad \text{and} \quad f(n) > a, \quad \text{then} \quad a \in A \leftrightarrow (\exists p < n)[a = F(p)],$$

and conclude that A is recursive in B. Suppose that A were truth-table reducible to B, say F and D are recursive and

$$a \in A \leftrightarrow \bar{\mathsf{K}}_B(F(a)) \in D.$$

Let G be a recursive function such that

$$\{G(m)\}(a) \quad \text{is defined iff} \quad \langle \mathsf{K}_B(0), \ldots, \mathsf{K}_B(m-1), 0, \ldots, 0 \rangle \notin D,$$

where the sequence on the right has length $F(a)$. Show that for every m, there is an $n \notin B$ such that $m \leqslant n < F(G(m))$ and conclude that A is recursive, a contradiction.)

5.10. Show that
(i) the class of open relations has the reduction property but not the separation property;
(ii) the class of closed relations has the separation property but not the reduction property (cf. Corollary 4.22 and Exercise 4.32).

5.11. Parallel to Definition 4.20, call a class X of relations *parametrizable* iff there exists a relation $\mathsf{U} \in X$ such that for every $\mathsf{R} \in X$, there exists a function β such that

$$\mathsf{R}(\mathbf{m}, \boldsymbol{\alpha}) \leftrightarrow \mathsf{U}(\langle \mathbf{m} \rangle, \langle \boldsymbol{\alpha} \rangle, \beta).$$

U is said to be *universal* for X. Show that the class of open relations is parametrizable.

5.12. Prove that the class of partial continuous functionals is the smallest class of partial functionals which includes ${}^{\omega}\omega$ and the class of primitive recursive functionals and is closed under expansion, functional composition, and unbounded search.

5.13. Show that if R is reducible to A, then R is recursive in A and if A is (semi-) recursive, then also R is (semi-) recursive. Give an example of R recursive in A but not reducible to A.

5.14. Show that there exists a primitive recursive function f such that if A is recursive in B with index a, then $A^{\omega J}$ is recursive in $B^{\omega J}$ with index $f(a)$.

5.15 Notes. The theory of degrees is one of the most developed areas of recursion theory. Shoenfield [1971a] is an excellent introduction. Many-one reducibility and other related notions are discussed in detail in Rogers [1967, Chapters 6–10].

Historically, the notion of relative recursiveness for functions on numbers preceded and led to the notion of recursive functional (see the first few pages of Kleene [1955b]). In our framework this would go as follows. First we would define for a fixed β a set $\Omega[\beta]$ of sequences of the form $(a, \mathbf{m}, n)$. The definition would be exactly like Definition 2.1 except that $\boldsymbol{\alpha}$ and the parameter l would be omitted and the clause introducing the initial functionals $\mathsf{Ap}^{k,l}$ would be replaced by one introducing β as an initial function:

$$(\langle 0, k, 3, i\rangle, \mathbf{m}, \beta(m_i)) \in \Omega[\beta].$$

We would write $\{a\}^{\beta}(\mathbf{m}) \simeq n$ iff $(a, \mathbf{m}, n) \in \Omega[\beta]$ and say F is partial recursive in β iff $F = \{a\}^{\beta}$ for some $a \in \omega$. Next we would extend this in the obvious way to $\Omega[\boldsymbol{\beta}]$, etc. and finally define F to be a partial recursive functional iff for some a, $\mathsf{F}(\mathbf{m}, \boldsymbol{\beta}) \simeq \{a\}^{\boldsymbol{\beta}}(\mathbf{m})$. Of course, in hindsight, this is a trivial variant, but it was not always so obvious.