MINIMAL LOGIC AND COMPUTERS

Chapter IV

Theory of Numbers in the System R

Part I

by

Frederic B. Fitch
Yale University

and

Richard J. Orgass
Rutgers University

Rutgers University
The State University of New Jersey
Department of Computer Science
New Brunswick, New Jersey 08903

Technical Report No. 22

CHAPTER IV*

THEORY OF NUMBERS IN THE SYSTEM R

---

* Earlier chapters were issued as IBM Research Reports by the Thomas J. Watson Research Center, Yorktown Heights, New York, 10598. These reports are:

Chapter I. The Method of Subordinate Proofs
RC-2503, June 9, 1969

Chapter II. Introductory Discussion
RC-2504, June 9, 1969

Chapter III. Natural Deduction Rules for the System R
RC-2754, January 12, 1970

† Sections marked with a daggar appear in this report; the remaining sections will appear in a subsequent report.

40. Introduction

Chapter IV contains a discussion of the general properties of the system $\underset{\sim}{R}$. In this chapter, the general properties of the system will be used for a detailed development of the theory of numbers in the system $\underset{\sim}{R}$.

Section 41 contains a detailed discussion of natural numbers and functions of natural numbers. The discussion begins by introducing the concept of an $\underset{\sim}{R}$-formula I-representing a natural number. Speaking informally, this is saying that certain R-formulas serve as names for natural numbers. This definition, together with the primitive rules for 'ω' are used to develop, in detail, the properties of addition, multiplication, exponentiation and proper subtraction in the system $\underset{\sim}{R}$.

Sections 42 and 43 are devoted to developing more general ways of dealing with functions of natural numbers in the system $\underset{\sim}{R}$. The basic idea is to introduce certain classes of functions of natural numbers and show that each function has an I-representation in system $\underset{\sim}{R}$. The particular classes of functions which are considered the primitive recursive functions and the partial recursive functions. Section 44 contains a discussion of general methods which can be used to show that relations among natural numbers have I-representations in the system $\underset{\sim}{R}$. There is also a discussion of limitations on relations which can be represented in the system $\underset{\sim}{R}$.

In section 45, the theory of natural numbers in the system $\underset{\sim}{R}$ is used to continue the development of the theory of combinators which was begun in section 31. Using natural numbers, it is possible to prove many additional properties of combinators as well as to define combinators which have interesting and useful properties.

Finally, in section 46, the concept of an ordinal number is introduced. Then it is shown that a substantial fragment of several versions of the theory of ordinal numbers can be developed within the system $\underset{\sim}{R}$.

In summary, the objective of this chapter is to develop the theory of natural numbers and ordinal numbers in the system $\underset{\sim}{R}$.

41. Natural Numbers

41.1 The theory of natural numbers will be developed in the system R using combinators. This will be done by letting the numerals '0', '1', '2', '3', and so forth serve as abbreviations for particular R-formulas which are built up out of 'B', 'C', 'I', 'W', and 'K'. The R-formulas which are abbreviated as numerals have the property that we can prove the following identities:

$$0\underline{ab} = \underline{b}$$

$$1\underline{ab} = \underline{ab}$$

$$2\underline{ab} = \underline{a}(\underline{ab})$$

$$3\underline{ab} = \underline{a}(\underline{a}(\underline{ab}))$$

and so forth. Speaking informally, as we were in Chapter III, we would say that these R-formulas denote the natural numbers 0, 1, 2, 3 and so forth.[1] However, we are now beginning the development of a more precise interpretation of certain R-formulas. In order to begin the development of this more precise interpretation, we will define the concept of an R-formula I-representing a natural number. The statement that an R-formula I-represents a natural number is a more precise way of saying that the R-formula denotes a natural number. The definition of an R-formula I-representing a natural number is stated in 41.2 and some examples are given in 41.3.

41.2 Definition. An R-formula '$\underline{a}$' is said to I-represent a natural number α just in the case that for all R-formulas '$\underline{x}$' and '$\underline{y}$', we can give a proof of

---

[1] Church [1] develops the theory of natural numbers in the calculi of lambda-conversion in this manner.

$$\underline{axy} = \underbrace{\underline{x}(\underline{x}(\underline{x}\ldots(\underline{xy})\ldots))}_{\alpha \text{ times}}$$

41.3 In exercise 30.26(4), the reader was asked to give a proof of the identity '$CK\underline{xy} = \underline{y}$'. Therefore, by definition 41.2, 'CK' I-represents the natural number 0. It is also possible to give a proof of '$I(CK)\underline{xy} = \underline{y}$'. Therefore, by definition 41.2, 'I(CK)' also I-represents the natural number 0. Similar remarks apply to 'I(I(CK))', 'I(I(I(CK)))', and so forth. It is straightforward to give a proof of the identity '$I\underline{xy} = \underline{xy}$'. Therefore, by definition 41.2, 'I' I-represents the natural number 1. The R-formulas 'II', 'III', 'IIII', and so forth also I-represent the natural number 1. Here is a proof of '$SB(CK)\underline{xy} = \underline{xy}$':

(*)	1	$SB(CK)\underline{xy} = B\underline{x}(CK\underline{x})\underline{y}$	S id
	2	$= \underline{x}(CK\underline{xy})$	1, B id
	3	$CK\underline{xy} = \underline{y}$	30.26(4)
	4	$SB(CK)\underline{xy} = \underline{xy}$	2,3, id elim

By definition 41.2, 'SB(CK)' also I-represents the natural number 1. In exercise 30.26(3), the reader was asked to give a proof of the identity '$WB\underline{xy} = \underline{x}(\underline{xy})$'. Therefore, by definition 41.2, 'WB' I-represents the natural number 2. The R-formulas 'I(WB)', 'I(I(WB))', and so forth also I-represent the natural number 2. Here is a proof of the identity '$SB(SB(CK))\underline{xy} = \underline{x}(\underline{xy})$':

	1	$SB(SB(CK))\underline{xy} = B\underline{x}(SB(CK)\underline{x})\underline{y}$	S id
	2	$= \underline{x}(SB(CK)\underline{xy})$	1, B id
	3	$SB(CK)\underline{xy} = \underline{xy}$	Proof (*)
	4	$SB(SB(CK))\underline{xy} = \underline{x}(\underline{xy})$	2,3, id elim

Therefore, by definition 41.2, 'SB(SB(CK))' also I-represents the natural number 2. Of course, 'I(SB(SB(CK)))', 'I(I(SB(SB(CK))))', 'I(I(I(SB(SB(CK)))))', and so forth all I-represent the natural number 2.

41.4 The symbol 'ω' is frequently used to denote the class of natural numbers. The class, ω, of natural numbers has the following properties: (1) The natural numbers 0 and 1 are in ω. (2) If a natural number, α, is in ω, then the successor of α, i.e., $\alpha+1$, is in ω. These properties of the class of natural numbers will now be used to show that each natural number has an I-representation. In 41.3 it was shown that 0 and 1 have I-representations. In order to show that each natural number has an I-representation, we must prove that $\alpha+1$ has an I-representation on the hypothesis that α has an I-representation. The statement that α has an I-representation is a more concise way of saying that there is an R-formula '$\underline{a}$' such that we can give a proof of:

(*) $\underline{axy} = \underbrace{\underline{x}(\underline{x}(\underline{x}\ldots(\underline{xy})\ldots))}_{\alpha \text{ times}}$

If we exhibit an R-formula '$\underline{b}$' such that we can give a proof of

(**) $\underline{baxy} = \underbrace{\underline{x}(\underline{x}(\underline{x}(\underline{x}\ldots(\underline{xy})\ldots)))}_{\alpha+1 \text{ times}}$

on the hypothesis (*), then we have shown that if there is an R-formula which I-represents α, then there is an R-formula which I-represents $\alpha+1$. Here is a proof of

(***) $\underline{SBaxy} = \underbrace{\underline{x}(\underline{x}(\underline{x}(\underline{x}\ldots(\underline{xy})\ldots)))}_{\alpha+1 \text{ times}}$

on the hypothesis (*):

1	$\vdash \underline{axy} = \underbrace{\underline{x}(\underline{x}(\underline{x}\ldots(\underline{xy})\ldots))}_{\alpha \text{ times}}$	hyp
2	$\underline{SBaxy} = \underline{Bx}(\underline{ax})\underline{y}$	S id
3	$= \underline{x}(\underline{axy})$	2, B id
4	$= \underbrace{\underline{x}(\underline{x}(\underline{x}(\underline{x}\ldots(\underline{xy})\ldots)))}_{\alpha+1 \text{ times}}$	1,3, id elim

Note that (***) was obtained from (**) by replacing '$\underline{b}$' in (**) with 'SB'. Thus, we have shown that if α has an I-representation, then $\alpha+1$ has an I-representation.

41.5 In 41.3 and 41.4 we outlined proofs of the following theorems.

41.6 Theorem. Every natural number has an I-representation.

41.7 Theorem. Every natural number has infinitely many I-representations. [Further, the class of R-formulas which I-represent a given natural number is countable.]

41.8 In 41.1 it was stated that we would let the numerals '0', '1', '2', '3', '4', and so forth serve as abbreviations for particular R-formulas which are built up out of combinators. The simplest way of stating these abbreviations is to say that '0' is an abbreviation for an R-formula which I-represents the natural number 0, '1' is an abbreviation for an R-formula which I-represents the natural number 1, and so forth. However, by Theorem 41.7, this statement would say that '0' is an abbreviation for any member of a countably infinite class of R-formulas. This is an unacceptable statement of an abbreviation. In order to avoid this difficulty,

we will let a numeral serve as an abbreviation for a particular R-formula which I-represents the natural number denoted by the numeral. The numerals which serve as abbreviations for R-formulas will be called R-numerals.

41.9 Definition.[2] The R-numerals are abbreviations for R-formulas which I-represent natural numbers. The R-numerals are defined by the following abbreviations:

'0' for 'CK'

'1' for 'SB0', that is, for 'SB(CK)',

'2' for 'SB1', that is, for 'SB(SB(CK))',

'3' for 'SB2', that is, for 'SB(SB(SB(CK)))',

'4' for 'SB3', that is, for 'SB(SB(SB(SB(CK))))',

and so forth. In the system R, a two-digit numeral such as twelve is designated by the expression '$\overline{12}$' to distinguish it from '12' which is the R-numeral '1' combined with the R-numeral '2'.

41.10 The reader should verify that it is possible to give a proof of:

$$\overline{12}\underline{xy} = \underbrace{\underline{x}(\underline{x}(\underline{x}\ldots(\underline{xy})\ldots))}_{12 \text{ times}}$$

and of:

$$12\underline{xy} = \underline{x}(\underline{xy})$$

41.11 The argument in 41.3 and 41.4 together with definition 41.9 amount to a proof of the identities for R-numerals. That is, we have given a proof of the basic identities for '0', '1', '2', and so forth. Hereafter, we will permit citing such identities

[2]The abbreviations used here are due to Curry [2].

as reasons in proofs. Here is a simple proof which uses such rules.

1	$0\underline{ab} = \underline{b}$	0 id
2	$1\underline{ab} = \underline{ab}$	1 id
3	$2\underline{ab} = \underline{a}(\underline{ab})$	2 id
4	$\overline{12}\underline{ab} = \underline{a}(\underline{a}(\underline{a}(\underline{a}(\underline{a}(\underline{a}(\underline{a}(\underline{a}(\underline{a}(\underline{a}(\underline{ab}))))))))))$	$\overline{12}$ id

The above statements amount to saying that we have a derived identity rule for each R-numeral. Of course, we can use these identities to derive introduction and elimination rules for R-numerals. Hereafter, we will use such rules. Here is a simple proof which uses 2 introduction and elimination:

1	$\underline{a}(\underline{ab})$	hyp	2, 2elim
2	$2\underline{ab}$	1, 2 int	hyp

41.12 In order to derive rules for arithmetic, it is desirable to have derived rules for the relation of equality available in the system R. We will use the symbol '$\doteq$' to denote this relation. A minimum requirement for this relation of equality is that we can give a proof of '$\underline{a} \doteq \underline{b}$' if '$\underline{a}$' and '$\underline{b}$' both I-represent the same natural number. By definition 41.2, if '$\underline{a}$' and '$\underline{b}$' both I-represent the same natural number, then we can give a proof of '$\forall_2\underline{xy}[\underline{axy} = \underline{bxy}]$'. In order to derive rules for the relation of equality among natural numbers, we will let '$\doteq$' serve as an abbreviation for '$\lambda_2\underline{ab}\forall_2\underline{xy}[\underline{axy} = \underline{bxy}]$'. Note that this definition of '$\doteq$' is such that it is not necessary to know that '$\underline{a}$' and/or '$\underline{b}$' I-represent natural numbers in order to give a proof of '$\underline{a} \doteq \underline{b}$'. It is straightforward to give proofs of the following rules for

equality; the proofs are left as exercises for the reader.

41.13 The derived rule of equality identity ($\doteq$ id). '$[\underline{a} \doteq \underline{b}] = \forall_2 \underline{xy}[\underline{axy} = \underline{bxy}]$' may appear as an item of any proof in the system R. Here is a simple proof which uses this rule:

1	$[\underline{a} \doteq \underline{b}] = \forall_2 \underline{xy}[\underline{axy} = \underline{bxy}]$	$\doteq$ id

41.14 The derived rule of equality introduction ($\doteq$ int). There are three forms of this rule. First form: '$\underline{a} \doteq \underline{b}$' is a consequence of '$\forall_2 \underline{xy}[\underline{axy} = \underline{bxy}]$'. Second form: '$\underline{a} \doteq \underline{b}$' is a consequence of a categorical subproof which is general for '$\underline{x}$' and '$\underline{y}$' and which has '$\underline{axy} = \underline{bxy}$' as an item. Third form: '$\underline{a} \doteq \underline{b}$' is a consequence of '$\underline{a} = \underline{b}$'. Here are simple proofs which use this rule:

1	$\vdash \forall_2 \underline{xy}[\underline{axy} = \underline{bxy}]$	hyp
2	$\underline{a} \doteq \underline{b}$	1, $\doteq$ int (first form)

1	x,y	...	...
.		.	.
.		.	.
.		.	.
i		$\underline{axy} = \underline{bxy}$	...
i+1	$\underline{a} \doteq \underline{b}$		1-i, $\doteq$ int (second form)

1	$\vdash \underline{a} = \underline{b}$	hyp
2	$\underline{a} \doteq \underline{b}$	1, $\doteq$ int (third form)

Hereafter, we will omit mentioning the particular form of this rule which we are using.

41.15 The derived rule of equality elimination (≐ elim). There are three forms of this rule. Two forms are stated here and the third form is stated in 41.67. First form: '$\forall_2 xy[axy = bxy]$' is a consequence of '$a \doteq b$'. Second form: '(...bxy...)' is a consequence of '(...axy...)' and '$a \doteq b$'. Here are simple proofs which use these forms of ≐ elim:

1	$a \doteq b$	hyp
2	$\forall_2 xy[axy = bxy]$	1, ≐ elim (first form)

1	$a \doteq b$	hyp
2	(...axy...)	hyp
3	(...bxy...)	1,2, ≐ elim (second form)

41.16 The derived left monotonic rule for equality (lft mon ≐). '$ca \doteq cb$' is a consequence of '$a \doteq b$'. Here is a proof of this rule:

1	$a \doteq b$		hyp
2	x,y	$a \doteq b$	1, reit
3		$\forall_2 xy[axy = bxy]$	2, ≐ elim
4		$axy = bxy$	3, u q elim
5		$caxy = cbxy$	4, mon id
6	$ca \doteq cb$		2-5, ≐ int

It does not appear to be possible to give a proof of a general monotonic rule for equality. The present form of this rule is sufficient for our purposes.

41.17 The derived rule of reflexivity of equality (refl ≐). '$a \doteq a$' may appear as an item of any proof in the system R. Here

is a simple proof which uses this rule:

1	$\underline{a} \doteq \underline{a}$	reit $\doteq$

41.18 The derived rule of symmetry of equality (sym $\doteq$). '$\underline{b} \doteq \underline{a}$' is a consequence of '$\underline{a} \doteq \underline{b}$'. Here is a simple proof which uses this rule:

1	$\underline{a} \doteq \underline{b}$	hyp
2	$\underline{b} \doteq \underline{a}$	1, sym $\doteq$

41.19 The derived rule of transitivity of equality (trans $\doteq$). '$\underline{a} \doteq \underline{c}$' is a consequence of '$\underline{a} \doteq \underline{b}$' and '$\underline{b} \doteq \underline{c}$'. Here is a simple proof which uses this rule:

1	$\underline{a} \doteq \underline{b}$	hyp
2	$\underline{b} \doteq \underline{c}$	hyp
3	$\underline{a} \doteq \underline{c}$	1,2, trans $\doteq$

41.20 The derived rule of excluded middle for equality (ex mid $\doteq$). '$[\underline{a} \doteq \underline{b}] \vee \neg[\underline{a} \doteq \underline{b}]$' may appear as an item of any proof in the system R. Here is a simple proof which uses this rule:

1	$[\underline{a} \doteq \underline{b}] \vee \neg[\underline{a} \doteq \underline{b}]$	ex mid $\doteq$

41.21 The relation of equality among natural numbers was introduced in order to facilitate the early stages of the development of arithmetic in the system R and to state rules for the primitive rules for the R-formula 'ω'. At first, our discussion of arithmetic in the system R will be quite informal. Later, in order to systematically develop the interpretation of certain R-formulas, we will introduce, in 43.__, the concept of an R-formula

I-representing a function. Speaking informally, an R-formula 'f' is said to I-represent a function, F, if we can give a proof of '$\underline{f}\underline{a} = \underline{b}$' just in the case that $F(\alpha) = \beta$ where '$\underline{a}$' and '$\underline{b}$' I-represent, respectively, α and β. The statement that a function, F, has an I-representation is a more precise way of saying that the system R can be used to prove that $F(\alpha_1, \ldots, \alpha_n) = \beta$, provided F has a value for $\alpha_1, \ldots, \alpha_n$ as arguments. Such a proof amounts to a computation of the value of F for $\alpha_1, \ldots, \alpha_n$ as arguments.

41.22 Convention. Hereafter, expressions of the form '$\underline{a}\ \underline{b}\ \underline{c} = \underline{d}\ \underline{e}\ \underline{f}$' and '$\underline{a}\ \underline{b}\ \underline{c} \doteq \underline{d}\ \underline{e}\ \underline{f}$' will be used as abbreviations for expressions of the form '$[\underline{a}\ \underline{b}\ \underline{c}] = [\underline{d}\ \underline{e}\ \underline{f}]$' and '$[\underline{a}\ \underline{b}\ \underline{c}] \doteq [\underline{d}\ \underline{e}\ \underline{f}]$', respectively. Similarly, for example, '$\underline{ab}\ \underline{c}\ \underline{de} = \underline{fg}\ \underline{h}\ \underline{ij}$' and '$\underline{ab}\ \underline{c}\ \underline{de} \doteq \underline{fg}\ \underline{h}\ \underline{ij}$' serve, respectively, as abbreviations for '$[(\underline{ab})\ \underline{c}\ (\underline{de})] = [(\underline{fg})\ \underline{h}\ (\underline{ij})]$' and '$[(\underline{ab})\ \underline{c}\ (\underline{de})] \doteq [(\underline{fg})\ \underline{h}\ (\underline{ij})]$'. The parentheses which occur in the latter expressions can, of course, be omitted using convention 23.10.

41.23 Some of the "laws of arithmetic" are valid for all R-formulas while others are valid only for R-formulas which I-represent natural numbers. We will begin by deriving rules for arithmetic which are valid for all R-formulas. Then, we will use the relation of equality to state the primitive rules for the R-formula 'ω' and use these primitive rules to derive rules for arithmetic which are valid only for R-formulas which I-represent natural numbers. We will show, in 41.66, that if '$\underline{a}$' and '$\underline{b}$' I-

represent natural numbers, then it is possible to give a proof of '$\underline{a} = \underline{b}$' on the hypothesis '$\underline{a} \doteq \underline{b}$'. Some of the rules for arithmetic which are valid for all R-formulas can be derived as identities while others can only be derived as equalities. The rules which can be derived as equalities for all R-formulas can be derived as identities for R-formulas which I-represent natural numbers. However, we will need these rules as equalities in order to show that '$\underline{a} = \underline{b}$' is a consequence of '$\underline{a} \doteq \underline{b}$' provided '$\underline{a}$' and '$\underline{b}$' I-represent natural numbers. Thus, we may view the introduction of the relation of equality as a device which is used to state primitive rules for 'ω' and to obtain certain rules for arithmetic which are valid for all R-formulas.

41.24 In order to derive rules for multiplication in the system R, we will let '$\circ$' serve as an abbreviation for 'B'. Here is a proof of '$1 \circ \underline{a} \doteq \underline{a}$':

(*)	1	x,y	$[1 \circ \underline{a}]\underline{xy} = \circ 1\underline{axy}$	id int, def
	2		$= B1\underline{axy}$	1, rep, def
	3		$= 1(\underline{ax})\underline{y}$	2, B id
	4		$= \underline{axy}$	3, 1 id
	5	$1 \circ \underline{a} \doteq \underline{a}$		1-4, $\doteq$ int

Obviously, all rules for B are valid for multiplication. When we use a rule for B but with the symbol '$\circ$' in place of 'B', we will replace 'B' by 'mult' in the reason. For example, this convention can be used to abbreviate proof (*) as follows:

(**)				
	1	x,y	$[1 \circ \underline{a}]\underline{xy} = \circ 1\underline{axy}$	id int, def
	2		$= 1(\underline{ax})\underline{y}$	1, mult id
	3		$= \underline{axy}$	2, 1 id
	4		$1 \circ \underline{a} \doteq \underline{a}$	1-3, $\doteq$ int

Using convention 23.10, we can further abbreviate proof (**) as follows:

1	x,y	$[1 \circ \underline{a}]\underline{xy} = 1(\underline{ax})\underline{y}$	mult id
2		$= \underline{axy}$	1, 1 id
3		$[1 \circ \underline{a}] \doteq \underline{a}$	1-2, $\doteq$ int

We have given a proof of the derived rule of left-hand multiplication by one equality (lft mult 1 $\doteq$): '$1 \circ \underline{a} \doteq \underline{a}$' may appear as an item of any proof in the system R. Note that '$\underline{a}$' is any R-formula. It is not possible to derive a rule of right-hand multiplication by one equality which applies to all R-formulas. A rule of right-hand multiplication by 1 is stated in 41.70, below.

41.25 The derived rule of left-hand multiplication by zero equality (lft mult 0 $\doteq$): '$0 \circ \underline{a} \doteq 0$' may appear as an item of any proof in the system R. It is not possible to give a proof of right-hand multiplication by zero for all R-formulas. However, it can be proved for the case where the multiplied entity is a natural number.

41.26 Here is a proof of the equality '$\underline{a} \circ [\underline{b} \circ \underline{c}] \doteq [\underline{a} \circ \underline{b}] \circ \underline{c}$':

1	x,y	$[\underline{a} \circ [\underline{b} \circ \underline{c}]]\underline{x}\underline{y} = \underline{a}([\underline{b} \circ \underline{c}]\underline{x})\underline{y}$	mult id, mon id
2		$= \underline{a}(\underline{b}(\underline{c}\underline{x}))\underline{y}$	1, mult id
3		$= [\underline{a} \circ \underline{b}](\underline{c}\underline{x})\underline{y}$	2, mult id
4		$= [[\underline{a} \circ \underline{b}] \circ \underline{c}]\underline{x}\underline{y}$	3, mult id
5	$\underline{a} \circ [\underline{b} \circ \underline{c}] \doteq [\underline{a} \circ \underline{b}] \circ \underline{c}$		1-4, $\doteq$ int

We have given a proof of the following rule:

41.27 The derived rule of associativity of multiplication equality (assoc mult $\doteq$): '$\underline{a} \circ [\underline{b} \circ \underline{c}] \doteq [\underline{a} \circ \underline{b}] \circ \underline{c}$' may appear as an item of any proof in the system R. Here is a simple proof which uses this rule:

1	$\underline{a} \circ [\underline{b} \circ \underline{c}] \doteq [\underline{a} \circ \underline{b}] \circ \underline{c}$	assoc mult

41.28 We will use the symbol '↑' to denote exponentiation. The notation '$\underline{a} \uparrow \underline{b}$' is more frequently written '$\underline{a}^{\underline{b}}$'. For reasons which will become more obvious later, we will not use the more common notation. In order to derive rules for exponentiation, we will let '↑' serve as an abbreviation for 'T'. Obviously, all rules for T are valid for exponentiation. When we use such rules for 'T' with the symbol '↑', we will replace 'T' in the reason by 'exp'. The following rules for exponentiation are valid for all R-formulas '$\underline{a}$', '$\underline{b}$', and '$\underline{c}$'.

41.29 The derived rule of exponentiation by zero equality (exp 0 $\doteq$): '$\underline{a} \uparrow 0 \doteq 1$' may appear as an item of any proof in the system R. Here is a proof of this rule:

1	x,y	$[\underline{a} \uparrow 0]\underline{xy} = 0\underline{axy}$	exp id
2		$= \underline{xy}$	1, 0 id
3		$= 1\underline{xy}$	2, 1 id
4		$\underline{a} \uparrow 0 \doteq 1$	1-3, $\doteq$ int

41.30 The derived rule of exponentiation by one (exp 1): '$\underline{a} \uparrow 1 = \underline{a}$' may appear as an item of any proof in the system R. The proof of this rule is left as an exercise for the reader.

41.31 The derived rule of multiplication of expenents (mult exp): '$\underline{a} \uparrow [\underline{b} \circ \underline{c}] = [\underline{a} \uparrow \underline{c}] \uparrow \underline{b}$' may appear as an item of any proof in the system R. Here is a proof of this rule:

1		$[\underline{a} \uparrow [\underline{b} \circ \underline{c}]] = [\underline{b} \circ \underline{c}]\underline{a}$	exp id
2		$= \underline{b}(\underline{ca})$	1, mult id
3		$= [(\underline{ca}) \uparrow \underline{b}]$	2, exp id
4		$= [[\underline{a} \uparrow \underline{c}] \uparrow \underline{b}]$	3, exp id

41.32 In order to derive rules for addition in the system R, we will let '+' serve as an abbreviation for '$\lambda_3\underline{xyz}[\underline{xz} \circ \underline{yz}]$'. This abbreviation was chosen so that the rule of addition of exponents is derivable. It is straightforward to prove the following basic rules for addition.

41.33 The derived rule of addition identity (add id): '$+\underline{abc} = [\underline{ac} \circ \underline{bc}]$' may appear as an item of any proof in the system R. Here is a simple proof which uses this rule:

1		$+\underline{abc} = [\underline{ac} \circ \underline{bc}]$	ad id

41.34 The derived rules of addition introduction and elimination (add int, add elim). Introduction: '$+\underline{abc}$' is a consequence

of 'ac ∘ bc'. Elimination: 'ac ∘ bc' is a consequence of '+abc'. Here is a simple proof which uses these rules:

1	+abc	hyp	2, add int
2	ac ∘ bc	1, add elim	hyp

41.35 The derived rule of addition of exponents (add exp). '[a ↑ [b + c]] = [[a ↑ b] ∘ [a ↑ c]]' may appear as an item of any proof in the system R. Here is a proof of this rule:

1	[a ↑ [b + c]] = [b + c]a	exp id
2	= [ba ∘ ca]	1, add id
3	= [[a ↑ b] ∘ ca]	2, exp id
4	= [[a ↑ b] ∘ [a ↑ c]]	3, exp id

41.36 The derived rule of left-hand addition of zero equality (lft add 0 ≐): '0 + a ≐ a' may appear as an item of any proof in the system R.

41.37 The derived rule of right-hand addition of zero equality (rt add 0 ≐): 'a + 0 ≐ a' may appear as an item of any proof in the system R.

41.38 The associative rule for addition is valid for all R-formulas, 'a', 'b', and 'c' as an equality. Here is a proof of this rule:

1	x,y	[a + [b + c]]xy = [ax ∘ [b + c]x]y	add id, mon id
2		= [ax ∘ [bx ∘ cx]]y	1, add id
3		= [[ax ∘ bx] ∘ cx]y	2, assoc mult ≐ elim
4		= [[a + b]x ∘ cx]y	3, add id
5		= [[a + b] + c]xy	4, add id
6	a + [b + c] ≐ [a + b] + c		1-4, ≐ int

The formal statement of this rule is omitted.

41.39 The rule of right-hand distribution of multiplication into addition is valid for all R-formulas as an equality. The corresponding left-hand rule can be proved for R-formulas which I-represent natural numbers.

1	x,y	$[[\underline{a} + \underline{b}] \circ \underline{c}]\underline{x}\underline{y} = [\underline{a} + \underline{b}](\underline{c}\underline{x})\underline{y}$	mult id
2		$= [\underline{a}(\underline{c}\underline{x}) \circ \underline{b}(\underline{c}\underline{x})]\underline{y}$	1, add id
3		$= [[\underline{a} \circ \underline{c}]\underline{x} \circ [\underline{b} \circ \underline{c}]\underline{x}]\underline{y}$	2, mult id (twice)
4		$= [[\underline{a} \circ \underline{c}] + [\underline{b} \circ \underline{c}]]\underline{x}\underline{y}$	3, add id
5		$[\underline{a} + \underline{b}] \circ \underline{c} \doteq [\underline{a} \circ \underline{c}] + [\underline{b} \circ \underline{c}]$	1-4, $\doteq$ int

41.40 This concludes our discussion of rules for arithmetic which are valid for all R-formulas. It is possible to derive additional rules using the rules which have been presented. However, these additional rules are not necessary for our purposes. We will now state the primitive rules for the R-formula 'ω' and then use these primitive rules to derive rules for arithmetic which are valid for R-formulas which I-represent natural numbers. These primitive rules were chosen so that 'ω' will I-represent the class of natural numbers and so that ω is closed under the relation of equality among natural numbers.

41.41 The rule of ω introduction (ω int). There are two forms of this rule. First form: '$\omega\alpha$' where 'α' is an R-numeral, may appear as an item of any proof in the system R. Second form: '$\omega\underline{b}$' is a direct consequence of '$\omega\underline{a}$' and '$\underline{a} \doteq \underline{b}$'. This rule allows us to give a proof of '$\omega\underline{a}$' just in the case that '$\underline{a}$' I-represents a natural number. Here are simple proofs which use this rule:

1	$\omega 0$	ω int (first form)
2	$\omega 1$	ω int (first form)
3	$\omega\overline{12}$	ω int (first form)
4	$\omega\overline{56}$	ω int (first form)

1	$\omega\underline{a}$	hyp
2	$\underline{a} \doteq \underline{b}$	hyp
3	$\omega\underline{b}$	1,2, ω int (second form)

Hereafter, we will omit mentioning the particular form of this rule which we are using.

41.42 The rule of excluded middle for ω (ex mid ω). '$\omega\underline{a} \vee \neg(\omega\underline{a})$' may appear as an item of any proof in the system R. Speaking informally, this rule amounts to assuming that either a given R-formula I-represents a natural number or it does not I-represent a natural number. Here is a simple proof which uses this rule:

1	$\omega\underline{a} \vee \neg(\omega\underline{a})$	ex mid ω

41.43 Using the rule of excluded middle for ω it is straightforward to derive the rules of indirect proof and negation introduction for ω. The proofs of these rules are left as an exercise for the reader.

41.44 The derived rule of indirect proof for ω (ind prf ω). '$\omega\underline{a}$' is a consequence of a subproof which has '$\neg(\omega\underline{a})$' as its only hypothesis and which contains '$\underline{b}$' and '$\neg\underline{b}$' as items. Here is a simple proof which uses this rule:

1	| |_ ¬(ω<u>a</u>)	hyp
.	| | .	.
.	| | .	.
.	| | .	.
	| | <u>b</u>	...
i	| | ¬<u>b</u>	...
i+1	| ω<u>a</u>	1-i, ind prf ω

41.45 <u>The derived rule of negation introduction for</u> ω (<u>neg int</u> ω). '¬(ω<u>a</u>)' is a consequence of a subproof which has 'ω<u>a</u>' as its only hypothesis and which contains '<u>b</u>' and '¬<u>b</u>' as items. Here is a simple proof which uses this rule:

1	| |_ ω<u>a</u>	hyp
.	| | .	.
.	| | .	.
.	| | .	.
	| | <u>b</u>	...
i	| | ¬<u>b</u>	...
i+1	| ¬(ω<u>a</u>)	1-i, neg int ω

41.46 <u>The rule of induction for</u> ω (<u>induc</u> ω) <u>or</u> ω <u>elimination</u> (ω <u>elim</u>). '(...<u>a</u>...)' is a direct consequence of the following items: (1) 'ω<u>a</u>', (2) '(...0...)', and (3) a subproof which is general for '<u>b</u>', which has '(...<u>b</u>...)' as its only hypothesis and which contains '(...[1 + <u>b</u>]...)' as an item. Assuming this rule amounts to assuming that mathematical induction is valid in the system R. This rule can be derived using certain non-constructive rules which will be described in Chapter VIII. Here is an outline of a proof which uses this rule:

1	ωa	hyp
2	(...0...)	hyp
3	b | (...b...)	hyp
.	.	.
.	.	.
.	.	.
i	(...[1 + b]...)	...
i+1	(...a...)	1,2,3-i, induc ω

Here is a simple proof which uses this rule:

(*)	1	ωa	hyp
	2	1 + 0 ≐ 1	rt add 0 ≐
	3	≐ 0 + 1	2, lft add 0 ≐, sym ≐
	4	b | 1 + b ≐ b + 1	hyp
	5	1 + [1 + b] ≐ 1 + [b + 1]	4, lft mon ≐
	6	≐ [1 + b] + 1	5, assoc add ≐, trans ≐
	7	1 + a ≐ a + 1	1,3,4-6, induc ω

41.47 In addition to the primitive rules for ω which we have assumed, it is desirable to have available a rule of closure of ω under successor. This rule asserts that 'ω[1 + a]' is a consequence of 'ωa'. It is not possible to give a proof of this rule inside the system R. However, in the metalanguage we can argue that this is a reasonable rule. If we have given a proof of 'ωa', then there is a natural number α such that we can give a proof, which is general for 'x' and 'y', of:

(*) $$axy = \underbrace{x(x(x\ldots(xy)\ldots))}_{\alpha \text{ times}}$$

It is straightforward to give a proof, which is general for '$\underline{x}$' and '$\underline{y}$' of:

$$+1\underline{a}\underline{x}\underline{y} = \underbrace{\underline{x}(\underline{x}(\underline{x}(\underline{x}\ldots(\underline{x}\underline{y})\ldots))}_{\alpha+1 \text{ times}}$$

on the hypothesis (*). Further, there is an R-numeral 'α' such that '$\underline{a} = \alpha$'. We can give a proof of '$+1\underline{a} = S\mathrm{B}\alpha$'. The R-formula '$S\mathrm{B}\alpha$' is an R-numeral by definition 41.9. Therefore, using the first form of ω int, we can give a proof of '$\omega(S\mathrm{B}\alpha)$'. By the second form of ω int, '$\omega[1 + \underline{a}]$' is a direct consequence of '$\omega(S\mathrm{B}\alpha)$' and '$+1\underline{a} = S\mathrm{B}\alpha$'. Since we cannot formalize this proof in the system R, we will add the following primitive rule:

41.48 The rule of closure of ω under successor (clos ω suc). '$\omega[1 + \underline{a}]$' is a consequence of '$\omega\underline{a}$'. Here is a simple proof which uses this rule:

1	$\vdash \omega\underline{a}$	hyp
2	$\omega[1 + \underline{a}]$	1, clos ω suc

*41.49 [This paragraph is more advanced and may be omitted at first reading.] Instead of assuming primitive rules for 'ω' we could obtain some rules for ω in the following manner. Using the method described in 6_._, let 'γ' serve as an abbreviation for an R-formula such that we can give a proof of '$\gamma\underline{a}$' just in the case we can give a proof of '$[\underline{a} = \mathrm{CK}] \vee \exists\underline{x}[\gamma\underline{x} \wedge [\underline{a} = S\mathrm{B}\underline{x}]]$'. Let '$\omega$' serve as an abbreviation for '$\lambda\underline{x}\exists\underline{y}[\gamma\underline{y} \wedge [\underline{x} = \underline{y}]]$'. Using this definition, it is straightforward to give a proof of the rules ω introduction and closure of ω for successor. It does not appear to be possible to prove '$\gamma\underline{a} \vee \neg(\gamma\underline{a})$' for all R-formulas '$\underline{a}$'. There-

fore, it is apparently impossible to prove the rule ex mid ω using the above definition of 'ω'. Further, using only constructive rules it appears to be impossible to derive the rule of induction for ω (41.46) using the above definition of 'ω'.

41.50 Using the rule of closure of ω for successor it is possible to derive a second form of the rule of induction for ω. This form of the rule permits 'ω$\underline{b}$' as a second hypothesis in the general proof. Here is a sketch of a proof which uses the second form of induction for ω:

(*)	1	ω$\underline{a}$	hyp
	2	(...0...)	hyp
	3	$\underline{b}$ | ω$\underline{b}$	hyp
	4	(...$\underline{b}$...)	hyp
	5	.	.
		.	.
	.	.	.
	.		
	.		
	j	(...[1 + $\underline{b}$]...)	...
	j+1	(...$\underline{a}$...)	1,2,3-j, induc ω

In order to give a proof of this rule, we observe that a proof of the form (*) can be converted into a proof of the following form

1	ω$\underline{a}$	hyp
2	(...0...)	hyp
3	ω0	ω int
4	(...0...) ∧ ω0	2,3, conj int
5	$\underline{b}$ | (...$\underline{b}$...) ∧ ω$\underline{b}$	hyp
6	ω$\underline{b}$	5, conj elim

7	(...b...)	5, conj elim
.	.	steps 5 to j of proof (*)
.	.	
.	.	
i	(...[1 + b]...)	
i+1	ω[1 + b]	6, clos ω suc
i+2	(...[1 + b]...) ∧ ω[1 + b]	i,(i+1), conj int
i+3	(...a...) ∧ ωa	1,4,5-(i+2), induc ω
i+4	(...a...)	i+3, conj elim

We have given a proof of the following rule:

41.51 The derived second form of the rule of induction for ω or ω elimination (induc ω or ω elim). '(...a...)' is a consequence of the following three items: (1) 'ωa', (2) '(...0...)', and (3) a subproof which is general for 'b' which has 'ωb' and '(...b...)' as its only hypotheses and which contains '(...[1 + b]...)' as an item. Proof (*) in 41.50 is a sketch of a proof which uses this form of induction for ω. Hereafter, we will not identify the particular form of this rule which we are using.

41.52 Proofs of rules of closure of ω for addition, multiplication, and exponentiation are given in 41.53 to 41.56. The statement of these rules should be obvious from the proof so these statements are omitted.

41.53 Closure of ω for addition (clos ω add).

1	ωa	hyp
2	ωb	hyp
3	0 + b ≐ b	lft add 0 ≐
4	ω[0 + b]	2,3, ω int

5	c	ω[c + b]	hyp
6		ω[1 + [c + b]]	5, clos ω suc
7		1 + [c + b] ≐ [1 + c] + b	assoc add ≐
8		ω[[1 + c] + b]	6,7, ω int
9		ω[a + b]	1,4,5-8, induc ω

41.54 Left-hand multiplication by successor equality (lft mult suc ≐).

1	x,y	[1 + a] ∘ b ≐ [1 ∘ b] + [a ∘ b]	rt dist mult add ≐
2		[[1 + a] ∘ b]xy = +[1 ∘ b][a ∘ b]xy	1, ≐ elim, u q elim
3		= [[1 ∘ b]x ∘ [a ∘ b]x]y	2, add id
4		= [1(bx) ∘ [a ∘ b]x]y	3, mult id
5		= 1(bx)([a ∘ b]xy)	4, mult id
6		= bx([a ∘ b]xy)	5, 1 id
7		= [bx ∘ [a ∘ b]x]y	6, mult id
8		= [b + [a ∘ b]]xy	7, add id
9		[1 + a] ∘ b ≐ b + [a ∘ b]	1-8, ≐ int

41.55 Closure of ω for multiplication (clos ω mult).

1		ωa	hyp
2		ωb	hyp
3		ω0	ω int
4		0 ∘ b ≐ b	lft mult 0
5		ω[0 ∘ b]	3,4, ω int
6	c	ω[c ∘ b]	hyp
7		ω[b +[c ∘ b]]	2,6, reit, clos ω add
8		b + [c ∘ b] ≐ [1 + b] ∘ c	lft mult suc ≐
9		ω[[1 + b] ∘ c]	7,8, ω int
10		ω[a ∘ b]	1,5,6-9, induc ω

41.56 Closure of ω for exponentiation (clos ω exp).

1	ω*a*	hyp
2	ω*b*	hyp
3	[*a* ↑ 0] ≐ 1	exp 0 ≐
4	ω1	ω int
5	ω[*a* ↑ 0]	3,4, ω int
6	*b* | ω[*a* ↑ *b*]	hyp
7	*a* ↑ [1 + *b*] ≐ [*a* ↑ 1] ∘ [*a* ↑ *b*]	add exp, ≐ int
8	*a* ↑ 1 ≐ *a*	exp 1, ≐ int
9	ω[*a* ↑ 1]	1,8, reit, ω int
10	ω[[*a* ↑ 1] ∘ [*a* ↑ *b*]]	6,9, clos ω mult
11	ω[*a* ↑ [1 + *b*]]	7,10, ω int
12	ω[*a* ↑ *c*]	2,5,6-11, induc ω

41.57 In order to prove additional rules concerning natural numbers, we will now introduce the concept of an ordered pair.[3] We call 'BCT*ab*' an ordered pair. We say that '*a*' is the first term of the ordered pair 'BCT*ab*', and that '*b*' is the second term of the ordered pair. We will derive rules for BCT which enables us to refer to the first and second terms of ordered pairs. For brevity, ordered pairs will simply be called pairs. The derived rules for BCT which are stated in 41.58 to 41.61 will be used below.

[3]This definition of an ordered pair is similar to the definition of an ordered 2-tuple given in 44.__. The two definitions are different in essential ways and we use different words to refer to the two concepts.

41.58 The derived rule of BCT identity (BCT id).

'BCTabc = cab' may appear as an item of any proof in the system R. Here is a proof of this rule:

1	BCTabc = C(Ta)bc	B id
2	= Tacb	1, C id
3	= cab	2, T id

41.59 The derived rule of BCT identity for K (BCT id K).

'BCTabK = a' may appear as an item of any proof in the system R. As a consequence of this rule, the first term of the pair 'BCTab' is the result of applying the pair to 'K'. Here is a simple proof which uses this rule:

1	BCTabK = a	BCT id K

The proof of this rule is left as an exercise for the reader.

41.60 The derived rule of BCT identity for zero (BCT id 0).

'BCTab0 = b' may appear as an item of any proof in the system R.[4] As a consequence of this rule, the second term of the pair 'BCTab' is the result of applying the pair to zero. Here is a simple proof which uses this rule:

1	BCTab0 = b	BCT id 0

The proof of this rule is left as an exercise for the reader.

41.61 The derived rule of identity for pairs (id prs).

There are two forms of this rule. First form: 'a = c' and 'b = d' are consequences of 'BCTab = BCTcd'. Second form:

---

[4] An alternate form of this rule is: 'BCTabc = b' is a consequence of 'c ≠ 0'.

'BCT$\underline{a}\underline{b}$ = BCT$\underline{c}\underline{d}$' is a consequence of '$\underline{a}$ = $\underline{c}$' and '$\underline{b}$ = $\underline{d}$'. Here is a simple proof which uses this rule:

1	BCT$\underline{a}\underline{b}$ = BCT$\underline{c}\underline{d}$	hyp	2,3, id prs (second form)
2	$\underline{a}$ = $\underline{c}$	1, id prs (first form)	hyp
3	$\underline{b}$ = $\underline{d}$	1, id prs (first form)	hyp

The proof of this rule is left as an exercise for the reader. Hereafter, we will omit mentioning the particular form of this rule we are using.

41.62 We will now define a combinator D which has the property that if D is applied to the pair BCT$\underline{a}\underline{b}$, the result is the pair BCT[1 + $\underline{a}$]$\underline{a}$. In order to derive rules for D, we let 'D' serve as an abbreviation for '$\lambda\underline{x}$(BCT[1 + $\underline{x}$K]($\underline{x}$K))'.

41.63 The derived rule of D identity (D id). 'D(BCT$\underline{a}\underline{b}$) = BCT[1 + $\underline{a}$]$\underline{a}$' may appear as an item of any proof in the system R. Here is a simple proof which uses this rule:

1	D(BCT$\underline{a}\underline{b}$) = BCT[1 + $\underline{a}$]$\underline{a}$	D id

41.64 Consider the effect of multiple applications of D to the pair BCT00. One application of D to BCT00 gives BCT10. A second application of D to BCT00, that is, an application of D to BCT10, gives BCT21. A third application of D to BCT00 gives BCT32. In fact, we can prove:

0D(BCT00) = BCT00,

1D(BCT00) = BCT10,

2D(BCT00) = BCT21,

3D(BCT00) = BCT32,

and so forth. This property of D is summarized in:

41.65 The derived rule of multiple applications of D (m D). '[1 + a]D(BCT00) = BCT[1 + a]a' is a consequence of 'ωa'. Here is a proof of this rule:

1	⊢	ωa	hyp
2		[1 + 0] ≐ 1	rt add 0
3		$\forall_2$xy[[1 + 0]xy = 1xy]	2, ≐ elim
4		[1 + 0]D(BCT00) = 1D(BCT00)	3, u q elim
5		= D(BCT00)	4, 1 id
6		= BCT[1 + 0]0	5, D id
7	b	⊢ [1 + b]D(BCT00) = BCT[1 + b]b	hyp
8		[1 + [1 + b]]D(BCT00) = [1D ∘ [1 + b]D](BCT00)	add id
9		= 1D([1 + b]D(BCT00))	8, mult id
10		= D([1 + b]D(BCT00))	9, 1 id
11		= D(BCT[1 + b]b)	7,10, id elim
12		= BCT[1 + [1 + b]]b	11, D id
13		[1 + a]D(BCT00) = BCT[1 + a]a	1,6,7-12, induc ω

41.66 We are now in a position to give a proof of 'a = b' on the hypotheses 'ωa', 'ωb', and 'a ≐ b'. We will use this rule in the remainder of our development of the theory of natural numbers. Here is the proof:

1	$\omega\underline{a}$	hyp
2	$\omega\underline{b}$	hyp
3	$\underline{a} \doteq \underline{b}$	hyp
4	$1 + \underline{a} \doteq 1 + \underline{b}$	3, lft mon $\doteq$
5	$\underline{a} = BCT[1 + \underline{a}]\underline{a}0$	BCT id 0
6	$= [1 + \underline{a}]D(BCT00)0$	1,5, m D
7	$= [1 + \underline{b}]D(BCT00)0$	4,6, $\doteq$ elim
8	$= BCT[1 + \underline{b}]\underline{b}0$	2,7, m D
9	$= \underline{b}$	8, BCT id 0

We have given a proof of the following rule:

41.67 The derived third form of the rule of equality elimination ($\doteq$ elim). '$\underline{a} = \underline{b}$' is a consequence of '$\omega\underline{a}$', '$\omega\underline{b}$', and '$\underline{a} \doteq \underline{b}$'. Here is a simple proof which uses this rule:

1	$\omega\underline{a}$	hyp
2	$\omega\underline{b}$	hyp
3	$\underline{a} \doteq \underline{b}$	hyp
4	$\underline{a} = \underline{b}$	1,2,3, $\doteq$ elim

41.68 The rules stated in 41.69 to 41.78 are derived using the third form of the rule of equality elimination and equalities which were previously derived.

41.69 The derived rule of left-hand multiplication by one (lft mult 1). '$1 \circ \underline{a} = \underline{a}$' is a consequence of '$\omega\underline{a}$'. Here is a proof of this rule:

1	⊢ ω$\underline{a}$	hyp
2	$\underline{1} \circ \underline{a} \doteq \underline{a}$	lft mult 1 ≐ (41.24)
3	ω$\underline{1}$	ω int
4	ω[$\underline{1} \circ \underline{a}$]	1,3, clos ω mult
5	$\underline{1} \circ \underline{a} = \underline{a}$	i.2,4, ≐ elim

41.70 The derived rule of right-hand multiplication by one (rt mult 1). '$\underline{a} \circ \underline{1} = \underline{a}$' is a consequence of 'ω$\underline{a}$'. Here is a simple proof which uses this rule:

1	⊢ ω$\underline{a}$	hyp
2	$\underline{a} \circ \underline{1} = \underline{a}$	1, rt mult 1

41.71 The derived rule of left-hand multiplication by zero (lft mult 0). '$\underline{0} \circ \underline{a} = \underline{0}$' is a consequence of 'ω$\underline{a}$'. Here is a simple proof which uses this rule:

1	⊢ ω$\underline{a}$	hyp
2	$\underline{0} \circ \underline{a} = \underline{0}$	1, lft mult 0

A proof of right-hand multiplication by zero is given in 41.81, below.

41.72 The derived rule of associativity of multiplication (assoc mult). '$\underline{a} \circ [\underline{b} \circ \underline{c}] = [\underline{a} \circ \underline{b}] \circ \underline{c}$' is a consequence of 'ω$\underline{a}$', 'ω$\underline{b}$', and 'ω$\underline{c}$'. Here is a proof of this rule:

1	ω$\underline{a}$	hyp
2	ω$\underline{b}$	hyp
3	⊢ ω$\underline{c}$	hyp
4	$\underline{a} \circ [\underline{b} \circ \underline{c}] \doteq [\underline{a} \circ \underline{b}] \circ \underline{c}$	assoc mult ≐
5	ω[$\underline{b} \circ \underline{c}$]	2,3, clos ω mult

6	$\omega[\underline{a} \circ [\underline{b} \circ \underline{c}]]$	1,5, clos ω mult
7	$\omega[\underline{a} \circ \underline{b}]$	1,2, clos ω mult
8	$\omega[[\underline{a} \circ \underline{b}] \circ \underline{c}]$	3,7, clos ω mult
9	$\underline{a} \circ [\underline{b} \circ \underline{c}] = [\underline{a} \circ \underline{b}] \circ \underline{c}$	4,6,8, $\doteq$ elim

41.73 The derived rule of exponentiation by zero (exp 0). '$\underline{a} \uparrow 0 = 1$' is a consequence of '$\omega\underline{a}$'. Here is a proof of this rule:

1	$\omega\underline{a}$	hyp
2	$\omega 0$	ω int
3	$\omega 1$	ω int
4	$\omega[\underline{a} \uparrow 0]$	1,2, clos ω exp
5	$\underline{a} \uparrow 0 \doteq 1$	exp 0 $\doteq$
6	$\underline{a} \uparrow 0 = 1$	3,4,5, $\doteq$ elim

41.74 The derived rule of left-hand addition of zero (lft add 0). '$0 + \underline{a} = \underline{a}$' is a consequence of '$\omega\underline{a}$'. Here is a simple proof which uses this rule:

1	$\omega\underline{a}$	hyp
2	$0 + \underline{a} = \underline{a}$	1, lft add 0

41.75 The derived rule of right-hand addition of zero (rt add 0). '$\underline{a} + 0 = \underline{a}$' is a consequence of '$\omega\underline{a}$'. Here is a simple proof which uses this rule:

1	$\omega\underline{a}$	hyp
2	$\underline{a} + 0 = \underline{a}$	1, rt add 0

41.76 The derived rule of associativity of addition (assoc add). '$\underline{a} + [\underline{b} + \underline{c}] = [\underline{a} + \underline{b}] + \underline{c}$' is a consequence of '$\omega\underline{a}$', '$\omega\underline{b}$',

and '$\omega\underline{c}$'. Here is a simple proof which uses this rule:

1	$\omega\underline{a}$	hyp
2	$\omega\underline{b}$	hyp
3	$\omega\underline{c}$	hyp
4	$\underline{a} + [\underline{b} + \underline{c}] = [\underline{a} + \underline{b}] + \underline{c}$	1,2,3, assoc add

41.77 The derived rule of right-hand distribution of multiplication into addition (rt dist mult add). '$[\underline{a} + \underline{b}] \circ \underline{c} = [\underline{a} \circ \underline{c}] + [\underline{b} \circ \underline{c}]$' is a consequence of '$\omega\underline{a}$', '$\omega\underline{b}$', and '$\omega\underline{c}$'. Here is a simple proof which uses this rule:

1	$\omega\underline{a}$	hyp
2	$\omega\underline{b}$	hyp
3	$\omega\underline{c}$	hyp
4	$[\underline{a} + \underline{b}] \circ \underline{c} = [\underline{a} \circ \underline{c}] + [\underline{b} \circ \underline{c}]$	1,2,3, rt dist mult add

A proof of the rule of left-hand distribution of multiplication into addition is given in 41.87.

41.78 The derived rule of left-hand multiplication by successor (lft mult suc). '$[1 + \underline{a}] \circ \underline{b} = \underline{b} + [\underline{a} \circ \underline{b}]$' is a consequence of '$\omega\underline{a}$' and '$\omega\underline{b}$'. Here is a proof of this rule:

1	$\omega\underline{a}$	hyp
2	$\omega\underline{b}$	hyp
3	$\omega[1 + \underline{a}]$	1, clos ω suc
4	$\omega[[1 + \underline{a}] \circ \underline{b}]$	2,3, clos ω mult
5	$\omega[\underline{a} \circ \underline{b}]$	1,2, clos ω mult
6	$\omega[\underline{b} + [\underline{a} \circ \underline{b}]$	2,5, clos ω add
7	$[1 + \underline{a}] \circ \underline{b} \doteq \underline{b} + [\underline{a} \circ \underline{b}]$	lft mult suc $\doteq$
8	$[1 + \underline{a}] \circ \underline{b} = \underline{b} + [\underline{a} \circ \underline{b}]$	4,6,7, $\doteq$ elim

41.79 Convention. Hereafter, in proofs which use the rules of closure of ω for successor, addition, multiplication, and exponentiation in order to derive identities which are valid for R-formulas which I-represent natural numbers we will omit these steps. In order to indicate that they have been omitted, we will add "clos" to the reason for a step which refers to this item and we will cite the appropriate step which asserts that certain R-formulas I-represent natural numbers. For example, using this convention, we would abbreviate the proof in 41.78 as follows:

1	$\omega\underline{a}$	hyp
2	$\_\ \omega\underline{b}$	hyp
3	$[1 + \underline{a}] \circ \underline{b} \doteq \underline{b} + [\underline{a} \circ \underline{b}]$	lft mult suc $\doteq$
4	$[1 + \underline{a}] \circ \underline{b} = \underline{b} + [\underline{a} \circ \underline{b}]$	1,2,3, $\doteq$ elim, clos

In addition, we will omit items of the form '$\omega 0$', '$\omega 1$', '$\omega 2$', and so forth. To indicate that such an item has been omitted, we will add "ω int" as a reason for a step which refers to such an omitted item. For example, the proof in 41.81 is an abbreviation for the following proof

1	$\_\ \omega\underline{a}$	hyp
2	$\omega 0$	ω int
3	$0 \circ 0 = 0$	2, lft mult 0
4	$\quad \omega\underline{b}$	hyp
5	$\quad \_\ \underline{b} \circ 0 = 0$	hyp
6	$\quad \omega 1$	ω int
7	$\quad [1 + \underline{b}] \circ [1 \circ 0] + [\underline{b} \circ 0]$	2,4,6, rt dist mult add, reit

8	| |	$= 0 + [\underline{b} \circ 0]$	2,6, lft mult 1, reit
9	| |	$= 0 + 0$	5,8, id elim
10	| |	$= 0$	2, lft add 0, reit
11	$\underline{a} \circ 0 = 0$		1,4-10, induc ω

41.80 In 41.81 to 41.88 we give proofs of additional rules for arithmetic. The proofs of these rules use rules which were introduced above. Since the statements of these rules are obvious, these statements are omitted.

41.81 Right-hand multiplication by zero (rt mult 0).

1	|_ $\omega\underline{a}$		hyp
2	$0 \circ 0 = 0$		lft mult 0, ω int
3	$\underline{b}$ | $\omega\underline{b}$		hyp
4	|_ $\underline{b} \circ 0 = 0$		hyp
5	$[1 + \underline{b}] \circ 0 = [1 \circ 0] + [\underline{b} \circ 0]$		4, rt dist mult add, clos
6		$= 0 + [\underline{b} \circ 0]$	5, lft mult 1, ω int
7		$= 0 + 0$	4,6, id elim
8		$= 0$	7, lft add 0, ω int
9	$\underline{a} \circ 0 = 0$		1,2,3-8, induc ω

41.82 Commutativity of addition of one (comm add 1).

1	|_ $\omega\underline{a}$		hyp
2	$1 + 0 = 1$		rt add 0, ω int
3		$= 0 + 1$	2, lft add 0, ω int
4	$\underline{b}$ | $\omega\underline{b}$		hyp

5		⊢ $1 + \underline{b} = \underline{b} + 1$	hyp
6		$1 + [1 + \underline{b}] = 1 + [1 + \underline{b}]$	5, mon id
7		$= [1 + \underline{b}] + 1$	4,6, assoc add, ω int
8	$1 + \underline{a} = \underline{a} + 1$		1,3,4-7, induc ω

41.83 Commutativity of addition (com add).

1	ω$\underline{a}$		hyp
2	⊢ ω$\underline{b}$		hyp
3	$0 + \underline{b} = \underline{b}$		2, lft add 0, ω int
4	$= \underline{b} + 0$		2,3, rt add 0, ω int
5	$\underline{c}$	ω$\underline{c}$	hyp
6		⊢ $\underline{c} + \underline{b} = \underline{b} + \underline{c}$	hyp
7		$[1 + \underline{c}] + \underline{b} = 1 + [\underline{c} + \underline{b}]$	2,5, assoc add, reit, ω int
8		$= 1 + [\underline{b} + \underline{c}]$	6,7, id elim
9		$= [1 + \underline{b}] + \underline{c}$	2,5,8, assoc add, reit
10		$= [\underline{b} + 1] + \underline{c}$	2,9, reit, com add 1
11		$= \underline{b} + [1 + \underline{c}]$	2,5,10, assoc add, reit, clos, ω int
12	$\underline{a} + \underline{b} = \underline{b} + \underline{a}$		1,4,5-10, induc ω

41.84 Commutativity of multiplication by one (comm mult 1).

1	⊢ ω$\underline{a}$		hyp
2	$\underline{a} \circ 1 = \underline{a}$		1, rt mult 1
3	$= 1 \circ \underline{a}$		1,2, lft mult 1

41.85 Commutative associativity of addition (com assoc add).

1	ωa	hyp
2	ωb	hyp
3	ωc	hyp
4	ωd	hyp
5	$[a + b] + [c + d] = [[a + b] + c] + d$	1,2,3,4, assoc add, clos
6	$= [a + [b + c]] + d$	1,2,3,5, assoc add
7	$= [a + [c + b]] + d$	2,3,6, com add
8	$= [[a + c] + b] + d$	1,2,3,7, assoc add
9	$= [a + c] + [b + d]$	1,2,3,4,8, assoc add, clos

41.86 Right-hand multiplication by successor (rt mult suc).

1		ωa	hyp
2		ωb	hyp
3		$0 \circ [1 + b] = 0 + 0$	2, lft mult 0, lft add 0, ω int
4		$= 0 + [0 \circ b]$	2,3, lft mult 0
5	c	ωc	hyp
6		$c \circ [1 + b] = c + [c \circ b]$	hyp
7		$[1 + c] \circ [1 + b] = [1 + b] + [c \circ [1 + b]]$	2,5, lft mult suc, clos
8		$= [1 + b] + [c + [c \circ b]]$	6,7, id elim
9		$= [1 + c] + [b + [c \circ b]]$	2,5,8, reit, com assoc add, clos
10		$= [1 + c] + [[1 + c] \circ b]]$	9,5,2, lft mult suc, clos
11		$a \circ [1 + b] = a + [a \circ b]$	1,4,5-10, induc ω

41.87 Commutativity of multiplication (com mult).

1	ωa	hyp
2	⌐ ωb	hyp
3	0 ∘ b = b ∘ 0	2, lft mult 0, rt mult 0, ω int
4	c | ωc	hyp
5	⌐ c ∘ b = b ∘ c	hyp
6	[1 + c] ∘ b = b + [c ∘ b]	2,4, lft mult suc
7	= b + [b ∘ c]	5,6, id elim
8	= b ∘ [1 + c]	2,4, reit, rt mult suc
9	[a ∘ b] = [b ∘ a]	1,3,4-8, induc ω

41.88 Left-hand distribution of multiplication into addition (lft dist mult add).

1	ωa	hyp
2	ωb	hyp
3	⌐ ωc	hyp
4	ω[b + c]	2,3, clos ω add
5	a ∘ [b + c] = [b + c] ∘ a	1,2,3,4, com mult
6	= [b ∘ a]+ [c ∘ a]	1,2,3,5, rt dist mult add, clos
7	= [a ∘ b] + [a ∘ c]	1,2,3, com mult (twice)

41.89 The rules for arithmetic which are described in 41.89 to 41.93 were derived with the help of the rules for D.

41.90 The derived rule of cancellation for successor (can suc). 'a = b' is a consequence of 'ωa', 'ωb', and '1 + a = 1 + b'. Here is a proof of this rule:

1	ω$\underline{a}$	hyp
2	ω$\underline{b}$	hyp
3	$1 + \underline{a} = 1 + \underline{b}$	hyp
4	$\underline{a} = BCT[1 + \underline{a}]\underline{a}0$	BCT id 0
5	$= [1 + \underline{a}]D(BCT00)0$	1,4, m D
6	$= [1 + \underline{b}]D(BCT00)0$	3,5, id elim
7	$= BCT[1 + \underline{b}]\underline{b}0$	6,2, m D
8	$= \underline{b}$	7, BCT id 0

41.91 The derived rule of monotony of successor with respect to non-identity (mon suc non-id). '$1 + \underline{a} \neq 1 + \underline{b}$' is a consequence of 'ω$\underline{a}$', 'ω$\underline{b}$' and '$\underline{a} \neq \underline{b}$'. Here is a proof of this rule:

1	ω$\underline{a}$	hyp
2	ω$\underline{b}$	hyp
3	$\underline{a} \neq \underline{b}$	hyp
4	$\quad 1 + \underline{a} = 1 + \underline{b}$	hyp
5	$\quad \underline{a} = \underline{b}$	1,2,4, reit, can suc
6	$\quad \underline{a} \neq \underline{b}$	3, reit
7	$1 + \underline{a} \neq 1 + \underline{b}$	4-6, neg int id

41.92 The derived rule of monotony of addition with respect to non-identity (mon add non-id). '$\underline{c} + \underline{a} \neq \underline{c} + \underline{b}$' is a consequence of 'ω$\underline{a}$', 'ω$\underline{b}$', 'ω$\underline{c}$', and '$\underline{a} \neq \underline{b}$'.

1	ωa	hyp
2	ωb	hyp
3	ωc	hyp
4	$a \neq b$	hyp
5	$0 + a = 0 + b$	hyp
6	$= b$	1,2,5, lft add 0, reit
7	a	1,2,6, lft add 0, reit
8	$a \neq b$	4, reit
9	$0 + a \neq 0 + b$	5-8, neg int id
10	d ωd	hyp
11	$d + a \neq d + b$	hyp
12	$[1 + d] + a = [1 + d] + b$	hyp
13	$= 1 + [d + b]$	2,10,12, assoc add, reit, ω int
14	$1 + [d + a] =$	1,10,13, assoc add, reit, ω int
15	$\omega[d + a]$	1,10, reit, clos ω add
16	$\omega[d + b]$	2,10, reit, clos ω add
17	$1 + [d + a] \neq 1 + [d + b]$	11,15,16, mon suc non-id, reit
18	$[1 + d] + a \neq [1 + d] + b$	12-18, neg int id
19	$c + a \neq c + b$	3,9,10-18, induc ω

41.93 The derived rule of cancellation for addition (can add).

'$a = b$' is a consequence of 'ωa', 'ωb', 'ωc', and '$c + a = c + b$'.

Here is a proof of this rule:

1	$\omega\underline{a}$	hyp
2	$\omega\underline{b}$	hyp
3	$\omega\underline{c}$	hyp
4	$\underline{c} + \underline{a} = \underline{c} + \underline{b}$	hyp
5	$\underline{a} \neq \underline{b}$	hyp
6	$\underline{c} + \underline{a} \neq \underline{c} + \underline{b}$	1,2,3,5, reit mon add non-id
7	$\underline{c} + \underline{a} = \underline{c} + \underline{b}$	4, reit
8	$\underline{a} = \underline{b}$	5-7, ind pr id

41.94 The derived rule of non-successorship of zero (non suc 0). '$0 \neq 1 + \underline{a}$' may appear as an item of any proof in the system R. Speaking informally, this non-identity states that 0 is not the successor of (is not 1 plus) any R-function.

1	$0 = 1 + \underline{a}$	hyp
2	$[0 \neq 0] = 00\neg[0 = 0]$	0 id, def
3	$= [1 + \underline{a}]0\neg[0 = 0]$	1,2, id elim
4	$= [10 \circ \underline{a}0]\neg[0 = 0]$	3, add id
5	$= 10(\underline{a}0\neg)[0 = 0]$	4, mult id
6	$= 0(\underline{a}0\neg)[0 = 0]$	5, 1 id
7	$= [0 = 0]$	6, 0 id
8	$0 = 0$	id int
9	$0 \neq 0$	7,8, id elim
10	$0 \neq 1 + \underline{a}$	1-9, neg int id

41.95 The class of natural numbers is not closed under subtraction. However, the class of natural numbers is closed under a kind of subtraction which is called <u>proper subtraction</u>. The symbol '$\dot{-}$' is usually used to denote proper subtraction. Speaking informally, proper subtraction is defined as follows:

$$x \mathbin{\dot{-}} y = \begin{cases} x - y & \text{if } x \geq y \\ 0 & \text{if } y \geq x \end{cases}$$

It will now be shown that there is an R-formula which I- represents proper subtraction. This will be done by letting 'P' serve as an abbreviation for an R-formula such that we can give a proof of '$P0 \doteq 0$' and a proof of '$P[1 + \underline{a}] \doteq \underline{a}$' on the hypothesis '$\omega\underline{a}$'. Then, '$\dot{-}$' will be defined as an abbreviation for '$\lambda\underline{xy}([P \uparrow \underline{y}]\underline{x})$' and this definition will be used to show that '$\dot{-}$' I-represents proper subtraction.

41.96 Let 'P' serve as an abbreviation for '$\lambda\underline{x}(\underline{x}D(BCT00)0)$'. Using this definition, it is possible to derive the rules for the predecessor function stated in 41.97 and 41.98. Some of the proofs of these rules are omitted and left as exercises for the reader.

41.97 <u>The derived rule of predeccesor identity (pred id)</u>. '$P[1 + \underline{a}] = \underline{a}$' is a consequence of '$\omega\underline{a}$'. Here is a simple proof which uses this rule:

1	$\omega\underline{a}$	hyp
2	$P[1 + \underline{a}] = \underline{a}$	$\vdash_7$ pred id

41.98 The derived rule of predecessor of zero identity (pred 0 id). There are two forms of this rule. First form: 'P0 = 0' may appear as an item of any proof in the system R. Second form: '[P ↑ a]0 = 0' is a consequence of 'ωa'. Here are simple proofs which use this rule:

1	P0 = 0	pred 0 id (first form)
1	ωa	
2	[P ↑ a]0 = 0	hyp 1, pred 0 id (second form)

The first form of this rule is a consequence of the second form. Here is a proof of the second form:

1		ωa	hyp
2		[P ↑ 0]0 = 0P0	exp id
3		= 0	2, 0 id
4	b	ωb	hyp
5		[P ↑ b]0 = 0	hyp
6		[P ↑ [1 + b]]0 = [1 + b]P0	exp id
7		= [1P ∘ bP]0	6, add id
8		= 1P(bP0)	7, mult id
9		= P(bP0)	8, 1 id
10		= P([P ↑ b]0)	9, exp id
11		= P0	6,10 id elim
12		= λx(xD(BCTOO)0)0	11, rep, def
13		= 0D(BCTOO)0	12, abs id
14		= BCTOOO	13, 0 id
15		= 0	14, BCT id 0
16	[P ↑ a]0 = 0		1,3,4-15, induc ω

41.99 In order to prove some additional properties of the predecessor function, it is convenient to use a successor function. Speaking informally, the successor function is defined by the identity:

$$\sigma(x) = 1 + x$$

In order to derive rules for the successor function in the system R, we will let 'σ' serve as an abbreviation for '$\lambda\underline{x}[1 + x]$'. It is straightforward to state and prove the rules successor introduction and elimination and successor identity. These rules will now be used to derive rules relating the predecessor and successor functions. These rules will, in turn, be used to derive rules for proper subtraction.

41.100 The derived rule of successor-addition identity (suc-add id). '$[\sigma \uparrow \underline{a}]\underline{b} = \underline{a} + \underline{b}$' is a consequence of '$\omega\underline{a}$' and '$\omega\underline{b}$'. Here is a proof of this rule:

1	$\omega\underline{a}$		hyp
2	$\omega\underline{b}$		hyp
3	$[\sigma \uparrow 0]\underline{b} = 0\sigma\underline{b}$		exp id
4	$= \underline{b}$		3, 0 id
5	$= b + 0$		2,4,rt add 0
6	$\underline{c}$	$\omega\underline{c}$	hyp
7		$[\sigma \uparrow \underline{c}]\underline{b} = \underline{b} + \underline{c}$	hyp
8		$[\sigma \uparrow [1 + \underline{c}]]\underline{b} = [[\sigma \uparrow 1] \circ [\sigma \uparrow \underline{c}]]\underline{b}$	add exp
9		$= [\sigma \circ [\sigma \uparrow \underline{c}]]b$	8, exp 1
10		$= \sigma([\sigma \uparrow \underline{c}]\underline{b}$	9, mult id
11		$= \sigma[\underline{b} + \underline{c}]$	7,10, id elim
12		$= 1 + [\underline{b} + \underline{c}]$	11, suc id
13		$= [1 + \underline{b}] + \underline{c}$	2,6,12, assoc add, reit, ω int
14	$[\sigma \uparrow \underline{a}]\underline{b} = \underline{b} + \underline{a}$		1,5,6-13, induc ω
15	$= \underline{a} + \underline{b}$		1,2,14, comm add

41.101 Both a predecessor and successor function have been defined. Are these two functions inverses of each other? The answer is only a qualified yes. If α and β are arbitrary natural numbers, then it is possible to prove '$[[P \uparrow \underline{a}] \circ [\sigma \uparrow \underline{a}]]\underline{b} = \underline{b}$'. However, the hypothesis '$\underline{a} \leq \underline{b}$' is required to prove '$[[\sigma \uparrow \underline{a}] \circ [P \uparrow \underline{a}]\underline{b} = \underline{b}$'. A proof of the first identity is given below. A proof of the second identity requires additional results which are obtained with the help of the first identity. The first form of the derived rule of predecessor successor identity (pred-suc id) is:
'$[[P \uparrow \underline{a}] \circ [\sigma \uparrow \underline{a}]]\underline{b} = \underline{b}$' is a consequence of '$\omega\underline{a}$'. Here is a proof of this rule:

1	$\omega\underline{a}$	hyp
2	$[[P \uparrow 0] \circ [\sigma \uparrow 0]]\underline{b} = [P \uparrow 0]([\sigma \uparrow 0]\underline{b})$	mult id
3	$= 0P(0\sigma)\underline{b}$	2, exp id
4	$= 0\sigma\underline{b}$	3, 0 id
5	$= \underline{b}$	4, 0 id
6	$\quad\omega\underline{m}$	hyp
7	$\quad[[P \uparrow \underline{m}] \circ [\sigma \uparrow \underline{m}]]\underline{b} = \underline{b}$	hyp
8	$\quad[[P \uparrow [1 + \underline{m}]] \circ [\sigma \uparrow [\underline{m} + 1]]]\underline{b}$	
	$\quad= [P \circ [P \uparrow \underline{m}] \circ [\sigma \uparrow \underline{m}] \circ \sigma]\underline{b}$	add exp, assoc mult
9	$\quad= [P \circ [P \uparrow \underline{m}] \circ [\sigma \uparrow \underline{m}]](\sigma\underline{b})$	8, mult id
10	$\quad= [P \circ [P \uparrow \underline{m}] \circ [\sigma \uparrow \underline{m}]][1 + \underline{b}]$	9, suc id
11	$\quad= P([[P \uparrow \underline{m}] \circ [\sigma \uparrow \underline{m}]][1 + \underline{b}])$	10, mult id
12	$\quad= P[1 + \underline{b}]$	7, 11, id elim
13	$\quad= \underline{b}$	12, pred id
14	$\quad[[P \uparrow [1 + \underline{m}]] \circ [\sigma \uparrow [1 + \underline{m}]]]\underline{b} = \underline{b}$	13, comm add 1
15	$[[P \uparrow \underline{a}] \circ [\sigma \uparrow \underline{a}]]\underline{b} = \underline{b}$	6-14, induc ω

41.102 With these preliminary results, it is possible to prove a number of properties of $\dot{-}$. By the definition of $\dot{-}$ and the second form of pred 0 id, we have a proof of: The derived rule of subtraction from zero (sub fr 0). '$[0 \dot{-} \underline{a}] = 0$' is a consequence of '$\omega\underline{a}$'. Here is a simple proof which uses this rule:

1	$\omega\underline{a}$	hyp
2	$[0 \dot{-} \underline{a}] = 0$	1, sub fr 0

41.103 The derived rule of subtraction of zero (sub 0). '$[\underline{a} \dot{-} 0] = \underline{a}$' is a consequence of '$\omega\underline{a}$'. The straightforward proof of this rule is left as an exercise for the reader. Here is a simple proof which uses this rule:

1	⊢	$\omega\underline{a}$	hyp
2	⊢	$[\underline{a} \dot{-} 0] = a$	1, sub 0

41.104 The derived rule of subtraction of self identity (sub self id). '$[\underline{a} \dot{-} \underline{a}] = 0$' is a consequence of '$\omega\underline{a}$'. Here is a simple proof which uses this rule:

1	⊢	$\omega\underline{a}$	hyp
2	⊢	$[\underline{a} \dot{-} \underline{a}] = 0$	1, sub self id

Here is a proof of this rule:

1	⊢	$\omega\underline{n}$	hyp
2	⊢	$[P \uparrow 0]0 = 0$	pred 0 id
3	$\underline{b}$ ⊢	$\omega\underline{b}$	hyp
4	⊢	$[P \uparrow \underline{b}] = 0$	hyp
5	⊢	$[P \uparrow [b + 1]] = [P \uparrow [\underline{b} + 1]](\sigma\underline{b})$	suc id
6	⊢	$= [[P \uparrow \underline{b}] \circ P](\sigma\underline{b})$	5, add exp
7	⊢	$= [P \uparrow \underline{b}](P(\sigma\underline{b}))$	6, mult id
8	⊢	$= [P \uparrow \underline{b}]([P \circ \sigma]\underline{b})$	7, mult id
9	⊢	$= [P \uparrow \underline{b}]\underline{b}$	8, pred-suc id
10	⊢	$= 0$	4,9, id elim
11	⊢	$[P \uparrow \underline{n}]\underline{n} = 0$	1,2,3-10, induc ω
12	⊢	$[\underline{n} \dot{-} \underline{n}] = 0$	11, rep, def

41.105 In order to give proofs of some of the properties of proper subtraction, it is desirable to have the relations "less than" and "less than or equal" among natural numbers available in the system R. These relations will be introduced by definition here; a detailed discussion of relations is given in section 44, below. '$\underline{a} \leq \underline{b}$' is an abbreviation

for '$\exists \underline{x}[\omega \underline{x} \wedge \underline{a} + \underline{x} = \underline{b}]$' and '$\underline{a} < \underline{b}$' is an abbreviation for '$[\underline{a} + 1] \leq \underline{b}$'. It is clear that these R-formulas have the required property. Furthermore, they obey the law of excluded middle.

41.106 The derived rule of subtraction identity (sub id). There are two forms of this rule. First form: '$[\underline{a} \dot{-} \underline{b}] = 0$' is a consequence of '$\omega \underline{a}$', '$\omega \underline{b}$' and '$\underline{a} \leq \underline{b}$'. Second form: '$[\underline{a} \dot{-} \underline{b}] + \underline{b} = \underline{a}$' is a consequence of '$\omega \underline{a}$', '$\omega \underline{b}$' and '$\underline{b} \leq \underline{a}$'. Here are simple proofs which use this rule:

1	$\omega \underline{a}$	hyp
2	$\omega \underline{b}$	hyp
3	$\underline{a} \leq \underline{b}$	hyp
4	$[\underline{a} \dot{-} \underline{b}] = 0$	1,2,3, sub id (first form)

1	$\omega \underline{a}$	hyp
2	$\omega \underline{b}$	hyp
3	$\underline{b} \leq \underline{a}$	hyp
4	$[\underline{a} \dot{-} \underline{b}] + \underline{b} = \underline{a}$	1,2,3, sub id

Here is a proof of the first form of sub id:

1		$\omega\underline{a}$	hyp
2		$\omega\underline{b}$	hyp
3		$\underline{a} \leq \underline{b}$	hyp
4		$\exists\underline{x}[\omega\underline{x} \wedge [\underline{x} + \underline{a} = \underline{b}]]$	3, rep, def (41.105)
5	$\underline{x}$	$\omega\underline{x} \wedge [\underline{x} + \underline{a} = \underline{b}]$	hyp
6		$\underline{x} + \underline{a} = \underline{b}$	5, conj elim
7		$[\underline{a} \dot{-} \underline{b}] = [P \uparrow \underline{b}]\underline{a}$	id int, def
8		$= [P \uparrow [\underline{x} + \underline{a}]]\underline{a}$	6,7, id elim
9		$= [[P \uparrow \underline{x}] \circ [P \uparrow \underline{a}]]\underline{a}$	8, add exp
10		$= [P \uparrow \underline{x}]([P \uparrow \underline{a}]\underline{a})$	9, mult id
11		$= [P \uparrow \underline{x}]0$	10, pred 0 id
12		$= 0$	11, pred 0 id
13		$[\underline{a} \dot{-} \underline{b}] = 0$	4,5-12, eq elim

Here is a proof of the second form of sub id:

1	$\omega\underline{a}$	hyp
2	$\omega\underline{b}$	hyp
3	$\underline{b} \leq \underline{a}$	hyp
4	$\exists\underline{x}[\omega\underline{x} \wedge [\underline{x} + \underline{b} = \underline{a}]]$	3, rep, def (41.105)
5	$\underline{x}$ | $\omega\underline{x} \wedge [\underline{x} + \underline{b} = \underline{a}]$	hyp
6	$\underline{x} + \underline{b} = \underline{a}$	5, conj elim
7	$[\underline{a} \dot{-} \underline{b}] = (P \uparrow \underline{b})\underline{a}$	id int, def
8	$= [P \uparrow \underline{b}][\underline{x} + \underline{b}]$	6,7, id elim
9	$= [P \uparrow \underline{b}]([\sigma \uparrow \underline{b}]\underline{x})$	8, suc-add id
10	$= [[P \uparrow \underline{b}] \circ [\sigma \uparrow \underline{b}]]\underline{x}$	9, mult id
11	$= \underline{x}$	10, pred-suc id
12	$[\underline{a} \dot{-} \underline{b}] + \underline{b} = \underline{a}$	6,11, id elim
13	$[\underline{a} \dot{-} \underline{b}] + \underline{b} = \underline{a}$	4,5-12, eq elim

41.107 Convention. As for addition, '$\underline{a} \dot{-} \underline{b} = \underline{c}$' will serve as an abbreviation for '$[\underline{a} \dot{-} \underline{b}] = \underline{c}$'. If a rule of the form '$[\underline{x} \circ \underline{y}]\underline{z} = \underline{x}$' has been established we will use it to replace '$\underline{x}(\underline{yz})$' with '$\underline{x}$' (or conversely) simply by citing the rule. This does not change the validity of the proof but will shorten several proofs. Also, to shorten proofs, if we have a formula of the form '$[\underline{x} \uparrow \underline{a} + \underline{b}]\underline{y}$' we will use the rule add exp to replace it with '$[\underline{x} \uparrow \underline{a}]([\underline{x} \uparrow \underline{b}]\underline{y})$' without intermediate steps; if either '$\underline{a}$' or '$\underline{b}$' or both are the numeral '1', the exponential will be omitted.

41.108 The second form of the derived rule of predecessor successor identity (pred-suc id). '$[[\sigma \uparrow \underline{a}] \circ (P \uparrow \underline{a})]\underline{b} = \underline{b}$' is a consequence of '$\omega\underline{a}$', '$\omega\underline{b}$' and '$\underline{a} \leq \underline{b}$'. Here is a simple proof which uses this rule:

1	$\underline{a}$	hyp
2	$\underline{b}$	hyp
3	$\underline{a}\ \ \underline{b}$	hyp
4	$[[\sigma \uparrow \underline{a}] \circ [P \uparrow a]]\underline{b} = \underline{b}$	1,2,3, pred-suc id (second form)

The proof of this rule, which uses the rule sub id, is left as an exercise for the reader.

41.109 The derived rule less than entails less than or equal for differences (< ent ≤ dif). '$1 \leq \underline{b} \dot{-} \underline{a}$' is a consequence of '$\omega\underline{a}$', '$\omega\underline{b}$', and '$\underline{a} < \underline{b}$'. Here is a simple proof which uses this rule:

1	$\omega\underline{a}$	hyp
2	$\omega\underline{b}$	hyp
3	$\underline{a} < \underline{b}$	hyp
4	$1 \leq \underline{b} \dot{-} \underline{a}$	1,2,3, < ent ≤ dif

Here is a proof of this rule:

1	$\omega\underline{a}$	hyp
2	$\omega\underline{b}$	hyp
3	$\underline{a} < \underline{b}$	hyp
4	$\underline{a} + 1 \leq \underline{b}$	3, def
5	$\exists\underline{x}[\omega\underline{x} \wedge [\underline{x} + [\underline{a} + 1] = \underline{b}]]$	4, def
6	$\underline{x}$ | $\omega\underline{x} \wedge [\underline{x} + [\underline{a} + 1] = \underline{b}]]$	hyp
7	$\underline{x} + [\underline{a} + 1] = \underline{b}$	6, conj elim
8	$[\sigma \uparrow [\underline{a} + 1]]\underline{x} = \underline{b}$	7, suc-add id
9	$[\sigma \uparrow \underline{a}](\sigma\underline{x}) = \underline{b}$	8, exp id
10	$[P \uparrow \underline{a}]([\sigma \uparrow \underline{a}](\sigma\underline{x})) = [P \uparrow \underline{a}]b$	9, lft mon id
11	$\sigma\underline{x} = [P \uparrow \underline{a}]\underline{b}$	10, pred-suc id
12	$\underline{x} + 1 = [P \uparrow \underline{a}]\underline{b}$	11, suc id
13	$\omega\underline{x}$	6, conj elim
14	$\omega\underline{x} \wedge [x + 1 = [P \uparrow \underline{a}]\underline{b}]$	12, 13 conj int
15	$\exists\underline{x}[\omega\underline{x} \wedge [\underline{x} + 1 = [P \uparrow \underline{a}]\underline{b}]$	14, eq int
16	$1 \leq [P \uparrow \underline{a}]\underline{b}$	15, def, rep
17	$1 \leq [P \uparrow \underline{a}]\underline{b}$	5, 6-16, eq elim
18	$1 \leq b \dot{-} a$	17, def, rep

41.110 The derived rule successor-subtraction identity (suc-sub id). '$\sigma[\underline{b} \dot{-} (\sigma\underline{a})] = [\underline{b} \dot{-} \underline{a}]$' is a consequence of '$\omega\underline{a}$', '$\omega\underline{b}$', and '$\underline{a} < \underline{b}$'. Here is a simple proof which uses this rule:

1	$\omega\underline{a}$	hyp
2	$\omega\underline{b}$	hyp
3	$\underline{a} < \underline{b}$	hyp
4	$\sigma[\underline{b} \dot{-} (\sigma\underline{a})] = [\underline{b} \dot{-} \underline{a}]$	1,2,3, suc-sub id

The proof of this rule, which uses < ent ≤ dif, suc id, add exp, lft mon id, and pred-suc id, is left as an exercise for the reader.

41.111 The derived rule of cancellation for subtraction (can sub). '$\underline{b} \dot{-} [\underline{b} \dot{-} \underline{a}] = \underline{a}$' is a consequence of '$\omega\underline{a}$', '$\omega\underline{b}$', and '$\underline{a} \leq \underline{b}$'. Here is a simple proof which uses this rule:

1	$\omega\underline{a}$	hyp
2	$\omega\underline{b}$	hyp
3	$\underline{a} \leq \underline{b}$	hyp
4	$\underline{b} \dot{-} [\underline{b} \dot{-} \underline{a}] = \underline{a}$	1,2,3, can sub

The proof of this rule uses the rule induc ω to prove '$[\underline{a} \leq \underline{b}] \rightarrow [\underline{b} \dot{-} [\underline{b} \dot{-} \underline{a}] = \underline{a}$' on the hypotheses '$\omega\underline{a}$' and '$\omega\underline{b}$'. Then, mp can be used to derive the rule.

1		ωa	hyp
2		ωb	hyp
3		$0 \leq b$	hyp
4		$b \dot{-} [b \dot{-} 0] = b \dot{-} [P \uparrow 0]b$	id int, def
5		$= [P \uparrow 0Pb]b$	4, def, exp id
6		$= 0PbPb$	5, exp id
7		$= [P \uparrow b]b$	6, 0 id, exp id
8		$= 0$	7, sub self id
9		$[0 \leq b] \rightarrow [b \dot{-} 0]] = 0]$	3-8, imp int, 41.105
10	x	ωx	hyp
11		$[x \leq b] \rightarrow [b \dot{-} [b \dot{-} x]] = x]$	hyp
12		$1 + x \leq b$	
13		$x < b$	12, eq int, def
14		$x \leq b$	13, 41.105
15		$b \dot{-} [b \dot{-} [1 + x]] = b \dot{-} \sigma[b \dot{-} x]$	13, suc-sub id
16		$= \sigma[b \dot{-} [b \dot{-} x]]$	13, 15, suc-sub id
17		$= \sigma x$	11, 14, reit, mp
18		$= 1 + x$	17, suc id
19		$[1 + x \leq b] \rightarrow [b \dot{-} [b \dot{-} [1 + x]] = 1 + x$	12-18, imp int, 41.105
20		$[a \leq b] \rightarrow [b \dot{-} [b \dot{-} a] = a]$	1,9,10-19, induc ω

41.112 This concludes the detailed development of arithmetic in the system R. In section 43, more powerful techniques for the development of the theory of numbers and functions of numbers will be introduced. Section 42 is an introduction to this development.

42. Introduction to Recursive Functions

42.1 Computations are performed by manipulating some "computing material." For example, the computing material might be symbols written on paper, beads on an abacus, gears in a calculating engine, electric currents in the circuits of a computing machine, and so forth. Obviously, the manipulations which are performed in a particular computation are selected because of a particular interpretation of the computing material. For example, consider the directions for adding two natural numbers, written in decimal notation, given in any elementary arithmetic book. The computing material, in this case, consists of strings of symbols written on paper. These strings are built up out of the symbols '0', '1', '2', '3', '4', '5', '6', '7', '8', and '9'. The particular directions for performing this addition were selected because the character '0' is interpreted as a name for the natural number zero, the character '1' as a name for the natural number one, and so forth. Also, the string '10' is interpreted as a name for the natural number ten, the string '11' as a name of the natural number eleven, and so forth.

42.2 Speaking informally, an algorithm is a complete finite set of directions for performing a calculation. That is, a complete set of directions for manipulating some computing material. These directions specify everything that is to be done and do not require any creative activity on the part of the individual who is performing the calculations. The set of directions for adding two numbers men-

tioned in 42.1 is an example of an algorithm. Note that a finite set of directions can give rise to finite or infinite computations. This is because instruction i+j might say "go back to step i."

42.3 As an example of a finite set of directions which leads to computations of finite or infinite length, consider the standard algorithm for computing the square root of a number written in decimal notation. If this set of directions is used to compute the square root of four, then the resulting computation is of finite length. On the other hand, if this set of directions is used to compute the square root of two, then the resulting computation is of infinite length. After each repetition of some of the instructions a more accurate approximation to the square root of two is obtained.

42.4 The concept of an algorithm does not depend on the particular computing material which is used to perform the computation. Clearly, if a set of directions for a computation using some particular computing material is available, then this set of directions can be modified to describe the same computation using a different computing material. This new set of directions may be more or less complicated than the original set of directions. Therefore, we may chose a convenient abstract model for the development of our theory. Consider the class of functions which have one or more natural numbers as arguments and which have natural numbers as values. We would like to divide this class of functions into two classes: (1) the class of functions such that for each function in this class there is an algorithm for computing the value of this function given its arguments and

(2) the class of functions such that there does not exist an algorithm for computing the value of this function given its arguments. The first of these classes is called the class of "computable functions."

42.5 One way to obtain these two classes of functions is to define a class of functions such that each member of this class is a computable function. Then, it must be shown that for each function which is not in this class there does not exist an algorithm for computing the value of this function given its arguments. In order to suggest some of the difficulties which are associated with the development of a theory of computable functions based on this approach, we will give an informal sketch of the historical development of this point of view. The concepts discussed in this section are treated more precisely in subsequent sections. In order to emphasize that this is an informal discussion, we will use ordinary functional notation in this section. It is important to keep in mind that this section is an introductory survey and that many interesting problems have been omitted.

42.6 A class of computable functions may be obtained in the following manner. Begin with a class of functions such that for each function in this class it is trivial to give an algorithm for computing the value of this function given its argument(s). Additional computable functions are obtained by combining these functions to form new functions such that the directions for computing the new functions are obtained by combining the directions for computing

the given functions. This amounts to a defintion of a class of functions by induction. The class of functions which we begin with is the basis of the induction. Each of the modes of combination which are allowed correspond to closure clauses of a definition by induction. An early attempt to define the class of computable functions by induction will now be described. In the early literature, this class of functions is called the class of "recursive functions" but in the current literature this class is called the class of "primitive recursive functions." The reason for these names is discussed below.

42.7 The basis class in the inductive definition of the class of primitive recursive functions contains the following functions which are clearly computable.

(1) The zero function, Z, which is defined by the identity $Z(\underline{x}) = 0$.

(2) The successor function, S, which is defined by the identity $S(\underline{x}) = 1 + \underline{x}$.

(3) The projection functions, that is, functions whose values are their i^{th} arguments. More specifically, the projection function U_i^m is defined by the following identity:

$$U_i^m(\underline{x}_1, \ldots, \underline{x}_m) = \underline{x}_i$$

for all $m \geq 1$ and all i such that $1 \leq i \leq m$.

42.8 The first of the two closure clauses in the definition of the class of primitive recursive functions states that this class of functions is closed under composition. For example, if

$\underline{f}$ and $\underline{g}$ are computable functions, then the function $\underline{h}$ defined by the following identity:

$$\underline{h}(\underline{x}) = \underline{f}(\underline{g}(\underline{x}))$$

is computable. It is easy to see that if there is a set of directions for computing $\underline{f}$ and $\underline{g}$, then these directions can be combined to give directions for computing $\underline{h}$. The general closure clause which is assumed is the following. If $\underline{f}$ is an n-ary primitive recursive function and if $g_1, \ldots, g_n$ are m-ary primitive recursive function, then the m-ary function $\underline{h}$ defined by the identity:

$$\underline{h}(\underline{x}_1, \ldots, \underline{x}_m) = \underline{f}(\underline{g}_1(\underline{x}_1, \ldots, \underline{x}_m), \ldots, \underline{g}_n(\underline{x}_1, \ldots, \underline{x}_m))$$

is a primitive recursive functions. It is easy to see that the directions for computing $\underline{f}, \underline{g}_1, \ldots, \underline{g}_n$ can be combined to give directions for computing $\underline{h}$.

42.9 The second closure clause in the definition of this class of functions is responsible for the name of these functions. It states that this class of functions is closed under primitive recursion. If a function is defined by recursion, then the directions for computing its value for n as an argument are given using the value of the function for $n-1$ as an argument and by giving directions for computing this function when this argument is zero. A simple case of such a definition is as follows:

$$\underline{f}(0) = n$$

$$\underline{f}(S(\underline{k})) = \underline{g}(\underline{k}, \underline{f}(\underline{k}))$$

where $\underline{g}$ is a binary function which is known to be primitive recursive. Example 42.10 indicates how such a definition is used to com-

pute values of a function which is defined in this manner. The closure clause which is assumed is that if g and h are, respectively, n-ary and (n+2)-ary primitive recursive functions, then the function f defined by primitive recursion as follows is an (n+1)-ary primitive recursive function:

$$f(x_1, \ldots, x_n, 0) = g(x_1, \ldots, x_n)$$

$$f(x_1, \ldots, x_n, S(k)) = h(x_1, \ldots, x_n, k, f(x_1, \ldots, x_n, k)).$$

42.10 Example. In order to illustrate the definition of a function by primitive recursion, we will give a definition of the sum function, s, which is informally defined by the identity $s(x, y) = x + y$. The function s is defined using the successor function, S, and two projection functions as follows:

$$s(x, 0) = U_1^1(x)$$

$$s(x, S(y)) = S(U_3^3(x, y, s(x, y))).$$

Note that the function which corresponds to g in 42.9 is U_1^1 and that the function which corresponds to h is defined by composition. It is necessary to use the projection functions so that the definition of s will be an instance of the closure clause described in 42.9. Using the above definition of s, the computation of $s(2, 3)$ proceeds as follows:

$$s(2, 3) = S(U_3^3(2, 3, s(2, 2)))$$

$$s(2, 2) = S(U_3^3(2, 2, s(2, 1)))$$

$$s(2, 1) = S(U_3^3(2, 1, s(2, 0)))$$

$$s(2, 0) = U_1^1(2) = 2.$$

Substituting in the above identities in the reverse order we have:

$$s(2, 1) = S(U_3^3(2, 1, 2))$$
$$= S(2)$$
$$= 3$$
$$s(2, 2) = S(U_3^3(2, 2, 3))$$
$$= S(3)$$
$$= 4$$
$$s(2, 3) = S(U_3^3(2, 3, 4))$$
$$= S(4)$$
$$= 5$$

This completes the computation of $s(2, 3)$.

42.11 For some time, it was believed that the class of functions described in 42.7 to 42.9 includes all computable functions. During this time, the class of functions which we just described was called the class of recursive functions. However, in 1928, Ackermann showed that there is a function which is not in the class of functions which we described and which is computable. This result indicated that the class of computable functions is larger than was previously believed. Ackermann showed that functions which are defined by means of $\underline{n}$ simultaneous recursions are computable. The particular function which Ackermann exhibited is called Ackermann's function and is defined as follows:

(i) $\underline{f}(0, \underline{y}) = S(\underline{y})$

(ii) $\underline{f}(S(\underline{x}), 0), = \underline{f}(\underline{x}, 1)$

(iii) $\underline{f}(S(\underline{x}), S(\underline{y})) = \underline{f}(\underline{x}, \underline{f}(S(\underline{x}), \underline{y}))$.

42.12 Example. Here is a computation of Ackermann's function for the arguments 1 and 1:

$\underline{f}(1, 1) = \underline{f}(0, \underline{f}(1, 0))$ by (iii)

$\underline{f}(1, 0) = \underline{f}(0, 1)$ by (ii)

$\underline{f}(0, 1) = 2$ by (i)

Substituting these identities in the reverse order, we have:

$\underline{f}(1, 0) = 2$

$\underline{f}(1, 1) = \underline{f}(0, 2)$

$= 3.$

Here is a computation of Ackermann's function for the arguments 2 and 2:

(1) $\underline{f}(2, 2) = \underline{f}(1, \underline{f}(2, 1))$ by (iii)

(2) $\underline{f}(2, 1) = \underline{f}(1, \underline{f}(2, 0))$ by (iii)

(3) $\underline{f}(2, 0) = \underline{f}(1, 1)$ by (ii)

(4) $\underline{f}(1, 1) = \underline{f}(0, \underline{f}(1, 0))$ by (iii)

(5) $\underline{f}(1, 0) = \underline{f}(0, 1)$ by (ii)

(6) $\underline{f}(0, 1) = 2$ by (i)

Substituting as indicated, we have:

(7) $\underline{f}(1, 0) = 2$ (6) in (5)

(8) $\underline{f}(1, 1) = \underline{f}(0, 2)$ (7) in (4)

(9) $= 3$ by (i)

(10) $\underline{f}(2, 0) = 3$ (9) in (3)

(11) $\underline{f}(2, 1) = \underline{f}(1, 3)$ (10) in (2)

Now, in order to proceed, we must compute $\underline{f}(1, 3)$. This is done as follows:

(12) $\underline{f}(1, 3) = \underline{f}(0, \underline{f}(1, 2))$ by (iii)

(13) $\underline{f}(1, 2) = \underline{f}(0, \underline{f}(1, 1))$ by (iii)

(14) $\underline{f}(1, 2) = \underline{f}(0, 3)$ (9) in (13)

(15) $= 4$ by (i)

Substituting (15) in (12) gives:

(16) $\underline{f}(1, 3) = \underline{f}(0, 4)$

(17) $= 5$ by (i)

Using these results, we obtain the following for $\underline{f}(2, 2)$:

(18) $\underline{f}(2, 1) = 5$ (17) in (11)

(19) $\underline{f}(2, 2) = \underline{f}(1, 5)$ (18) in (1)

In order to obtain the value of $\underline{f}(2, 2)$ we must compute the value of $\underline{f}(1, 5)$. This is done as follows:

(20) $\underline{f}(1, 5) = \underline{f}(0, \underline{f}(1, 4))$ by (iii)

(21) $\underline{f}(1, 4) = \underline{f}(0, \underline{f}(1, 3))$ by (iii)

Substituting (17) in (21) and using definition (i), we have:

(22) $\underline{f}(1, 4) = \underline{f}(0, 5)$

(23) $= 6$

Substituting (23) in (20) and using definition (i), we have:

(24) $\underline{f}(1, 5) = \underline{f}(0, 6)$

(25) $= 7$

Lastly, substituting (25) in (19), we have:

(26) $\underline{f}(2, 2) = 7.$

This completes the computation of $\underline{f}(2, 2)$. The reader should observe that there are two recursions used to compute values of Ackermann's function.

42.13 Ackermann's result clearly indicated that the definition of the class of computable functions which was in current use was in-

adequate. It was clear that some new definition was required. This new definition must, at least, include all functions which were already in the class of functions which were known to be computable as well as all functions which are defined by simultaneous recursion. In 1936, Kleene proposed an additional closure clause in the definition of the class of computable functions and showed that if this clause is added, then the resulting class of functions includes all functions defined by simultaneous recursion as well as additional functions.

42.14 The closure clause added by Kleene amounts to saying that if $\underline{f}$ is an $(n+1)$-ary recursive function and if there is a number $\underline{y}$ such that $\underline{f}(\underline{x}_1, \ldots, \underline{x}_n, \underline{y}) = 0$, then the n-ary function $\underline{g}$ whose value for $\underline{x}_1, \ldots, \underline{x}_n$ as arguments is the least natural number $\underline{y}$ such that $\underline{f}(\underline{x}_1, \ldots, \underline{x}_n, \underline{y}) = 0$ is in the class of recursive functions. If a function, $\underline{g}$, is defined in this manner, then $\underline{g}$ is said to be defined by "minimalization." Such a definition is stated by saying that $\underline{g}$ is defined by the following identity: $\underline{g}(\underline{x}_1, \ldots, \underline{x}_n) = \mu\underline{y}[\underline{f}(\underline{x}_1, \ldots, \underline{x}_n, \underline{y}) = 0]$ and 'μ' is called the "least number operator."

42.15 Example. It is straightforward to give a definition, by primitive recursion, of proper subtraction. This is left to the reader as an exercise. It was shown in example 42.10 that addition can be defined by primitive recursion. Suppose that the function $\underline{f}$ is defined by the following identity: $\underline{f}(\underline{x}, \underline{y}) = (\underline{x} + 3) \dot{-} \underline{y}$. It is straightforward to verify that $\underline{f}$ is a binary primitive recursive function. This is left to the reader as an exercise. Suppose the

function g is a unary function defined by the following identity: $g(x) = \mu y[f(x, y) = 0]$. Suppose we wish to compute $g(1)$. We would proceed as follows. First we would let y be zero:

$$f(1, 0) = (1 + 3) \dot{-} 0$$
$$= 4.$$

Since this is not the correct value of y we increment y by one until we obtain the correct value.

$$f(1, 1) = (1 + 3) \dot{-} 1$$
$$= 3$$
$$f(1, 2) = (1 + 3) \dot{-} 2$$
$$= 2$$
$$f(1, 3) = (1 + 3) \dot{-} 3$$
$$= 1$$
$$f(1, 4) = (1 + 3) \dot{-} 4$$
$$= 0.$$

Therefore, $g(1) = 4$.

42.16 A still wider class of functions than the class of computable or recursive functions is obtained if in Kleene's closure clause we omit the phrase "and if there is a number y such that $f(x_1, \ldots, x_n) = 0$". This wider class of functions is called the class of partial recursive functions. A function which does not have a value for some values of its argument is said to be a "partial function" and a function which has a value for all values of its arguments is said to be a "total function." All the members of the class of recursive functions are total functions but some members of the class of partial recursive functions are partial functions.

42.17 Example. We will define a partial recursive function which has a value for some arguments and which does not have a value for other arguments. Recall that addition and subtraction can be defined by primitive recursion. Suppose that f is a binary function which is defined by the following identity: $f(x, y) = (x \dot{-} 1) + y$. Let g be the unary function which is defined by the following identity: $g(x) = \mu y[f(x, y) = 0]$. Suppose we wish to compute $g(0)$. We proceed as follows:

$$f(0, 0) = (0 \dot{-} 1) + 0$$

$$= 0 \quad \text{since} \quad 0 \dot{-} 1 = 0 \quad \text{and} \quad 0 + 0 = 0.$$

Therefore, $g(0) = 0$. In a similar way it can be shown that $g(1) = 0$. Now suppose we wish to compute $g(2)$. We proceed as follows:

$$f(2, 0) = (2 \dot{-} 1) + 0 = 1$$

$$f(2, 1) = (2 \dot{-} 1) + 1 = 2$$

$$f(2, 2) = (2 \dot{-} 1) + 2 = 3$$

It is easy to see that this computation will continue indefinitely. This is because there is no natural number y such that when it is added to a positive integer the result is zero. There is, of course, an integer which has this property.

42.18 Intuitively, the class of partial recursive functions appears to be the class of functions such that for each function in this class there is an algorithm for computing the value of the function, if there is one, given the arguments; and, if a function is not in this class, then there does not exist an algorithm for computing the value of this function. Since this is a particularly important

statement, we would like to give a proof of this assertion. In view of the fact that it was once thought that this statement is true of the class of primitive recursive functions, a proof is particularly desirable. We would like to be sure that there is not another result similar to Ackermann's for the class of partial recursive functions. Unfortunately, it is not possible to give a proof of this statement. The reason for this is that we do not have a precise definition of the concept of an algorithm. The informal statements at the beginning of this section are not sufficiently precise to be used in the proof of a theorem. Further, there is no known precise formulation of the concept of an algorithm. What should be done about this situation?

42.19 In 1937 Turing published a paper in which he described a particularly simple automaton. He showed that the class of functions which can be computed by such a device is just the class of partial recursive functions. In 1936 Post published a paper in which he proposed an alternate definition of the class of partially computable functions. His approach is completely independent of the approach which we described in this section and is also independent of Turing's work. He also obtained the class of partial recursive functions. In 1936 Church published a paper in which he showed that the class of functions which can be defined in the various calculi of lambda-convension is just the class of partial recursive functions. The lambda-calculi are also independent of the approaches which we have mentioned. Based on this evidence, Church stated that it was his <u>opinion</u> that the concept of an algorithm corresponds exactly to the concept of a partial

recursive function. This statement is called Church's thesis. It is important to remember that this is as assumption and that it is not a theorem. A number of writers have criticized Church's thesis. However, this is not an appropriate place for a philosophical discussion of Church's thesis.

42.20 This section is to serve as an informal introduction to the more formal discussion of the class of partial recursive functions given in Section 43.

INDEX TO ABBREVIATIONS FOR RULES

SYSTEM R

A

B

C

Index to Abbreviations for Rules (System R) - Page 2

C (continued)

Index to Abbreviations for Rules (System R) - Page 3

D

E

F

G

Index to Abbreviations for Rules (System R) - Page 4

G (continued)

H

I

Index to Abbreviations for Rules (System R) - Page 5

I (continued)

J

K

L

M

Index to Abbreviations for Rules (System R) - Page 6

M (continued)

m p	33.4
m p c	33.11
m t p	32.33 (11)
mon add non-id	41.92
mon id	30.8
mon suc non-id	41.91
mult exp	41.31

N

neg abs elim	34.13
neg abs int	34.12
neg conj elim	32.26
neg conj int	32.27
neg dis elim	32.28
neg dis int	32.29
neg elim	32.13
neg_2 elim	32.12
neg exist elim	36.19
neg exist int	36.20
neg_2 int	32.11
neg int id	32.19
neg int ω	41.45
neg m abs elim	34.31
neg m abs int	34.30
neg univ elim	36.17
neg univ int	36.16

Index to Abbreviations for Rules (System R) - Page 7

N (continued)

Index to Abbreviations for Rules (System R) - Page 8

R (continued)

S

T

Index to Abbreviations for Rules (System R) - Page 9

U

V

W

XYZ

Index to Abbreviations for Rules (System R) - Page 10

XYZ (continued)

$\doteq$ elim	41.14, 41.67
$\doteq$ id	41.13
$\doteq$ int	41.14
$\cap$ elim	37.5
$\cap$ id	37.3
$\cap$ int	37.4
$\cup$ elim	37.9
$\cup$ id	37.7
$\cup$ int	37.8
$\subseteq$ elim	37.20
$\subseteq$ id	37.18
$\subseteq$ int	37.22
$\subset$ elim	37.31
$\subset$ id	37.29
$\subset$ int	37.30
$<$ def	41.105
$\leq$ def	41.105
$<$ ent $\leq$ dif	41.109