Peano Commutativity in $[S']$

Martin Røpcke

3rd July 2005

Contents

1 Introduction

This report is turned in for credit for the course “Logic” at DIKU summer 2005. In brief, the assignment is to prove the commutativity of addition within peano arithmetic using the Logiweb system developed by Klaus Grue. Logiweb will be introduced shortly. That is, in this report we shall carry out a proof of the proposition that for all natural numbers the following holds:

$$\mathrm{t} + \mathrm{r} = \mathrm{r} + \mathrm{t}, \tag{1}$$

and we shall use Klaus Grue’s Logiweb system to check our proof. Cf. [1]. However, we use $[\dot{x}]$ and $[\dot{y}]$ as our variables.

During the course we have been introduced to mathematical logic using the textbook *Introduction to Mathematical Logic* by Elliot Mendelson, cf. [4]. In order to get the full of this report we thus assume an elementary knowledge of logic as it is introduced in this book. However, we do give short introductions to some basic topics—Logiweb as well as logic—making this report selfcontained—at least to a graduate student of computer science at DIKU.

The report has been written from June 23rd to July 3rd. Due to this short period of time the proof is carried out as simply as possible discarding any fancy solutions even before they have been fully conceived of. The commutativity of addition is part of Proposition 3.2 in [4], p. 156, which is proved immediately afterwards on pp. 157f. We follow this proof backchaining from the actual proof of (1) through other results of the proposition and any necessary lemmas and theorems contained in the book. Besides Mendelson’s propositions we only make use of two tautologies and one lemma in the final proof.

We hope our intentions have been reached and that this report carries just a little of the fun it has been working with Logiweb. As Klaus always invites the Logiweb user to we do also: “Have fun”.

2 The Logiweb System

In this section we give a short introduction to the Logiweb system, a web-based system for distribution of mathematical definitions, lemmas, and proofs writing web-pages. Cf. [1]. For further details and elaboration on the myriad of intricacies and topics concerning the system we refer to Klaus Grue’s introduction to Logiweb, namely the check page. Alternatively, we recommend the peano page introducing peano arithmetic, which well displays the system “at work”. In order to obtain a thorough knowledge of the system one is advised to follow the course “Logic” or to read the base page (cf. [2]), but probably the best choice would be to undertake the exercise of writing a page proving some proposition of choice from [4] using Logiweb. After such an exercise any work in Logiweb and the hardship of doing it will most likely depend on the level of mathematics one is working with.

2.1 References

As hinted at above, Logiweb is a system in which users worldwide can distribute their own mathematical work and reuse the work of others. Mathematics in Logiweb is published by submitting a *Logiweb page* to the Logiweb system, that is, to a *Logiweb server*. Abstractly speaking, in a Logiweb page one can define a mathematical theory or system[1] and a language of this system within which one can then define, state, and prove mathematical lemmas, theorems, propositions, etc. A theory must be defined in one page. The proofs within a theory will be proof checked upon submission to the Logiweb system (actually they will be proof checked when compiling the page locally) and one then knows if the contents of a page are correct within the theory. A page can refer to other Logiweb pages in which a needed theory, results, functionalities etc. are defined. Thus, Logiweb works much like the world wide web where pages refer to each other using hyperlinks. The mesh of Logiweb servers translate Logiweb references to http urls making sure a page can find its references and that together they form a directed acyclic graph from top (the base) to bottom (a Logiweb page).

Essentially, the Logiweb reference of a page is a global hash key computed on the basis of the contents of the specific page. This ensures that each page has a unique key, that is, reference. The reference of this page in the special *kana*[2] format is:

```
nani
susu sina sasa neta saku  tinu sana kete susa niki
sisi suna tuka kana tasa  nike tuni tetu kiki sune
keti kuta ketu kesi kena  sasu sika nasa natu
```

`Kana` is a format developed for Logiweb making it possible to speak or pronounce a reference. This is done because Logiweb has been developed with the possibility of using a microphone as input tool in mind.

Also, one can refer to this page using the well-known http url:

```
http://www.diku.dk/~grue/logiweb/20050502/home/mrmr/peano-commutativity/latest/vector/
page.lgw
```

The two last ways to refer to a page is using the reference expressed in so-called *mixed endian hexadecimal*—which the kana reference is based on—or using a decimal reference. Cf. [1] and the `Reference` section in the source or `pyk` code in a Logiweb page. The reference section—known as the `BIBLIOGRAPHY`—of a Logiweb page will be introduced shortly.

[1] We shall use *theory* and *system* interchangeably, as is the case with *Logiweb system* and *Logiweb*.

[2] For details on kana we refer to http://yoa.dk/logiweb/doc/misc/kana.html.

2.2 The Base Page

Now, in the previous section we mentioned *the base* of the mesh of Logiweb pages. This is a certain part of Logiweb, which implements the basic and necessary functionalities of the mathematical part of the Logiweb system. That is, the *base page*—as it is denoted—defines the proof checker and most of what makes the mathematics of Logiweb possible. Actually, there can be more than one base page, but the one Klaus Grue has written will probably be *the* base page to begin with. But one is free to write a base page making up a different (or a similar) system. A base page is a page without references to other pages and thus forms the base of a mathematical system within Logiweb. For more on this we refer to the base page, Section 1.4 in particular.

We now know the necessary basics of the Logiweb system. Next, we take a closer look at the structure of a Logiweb page.

3 A Logiweb Page

In this section we briefly describe the most important details of a Logiweb page. That is, we introduce the Logiweb page, the language used in Logiweb, and the predefined structure all pages must be written according to. This introduction is adequate for getting started with Logiweb, but is in no manner satisfactory to the serious user of the Logiweb system.

3.1 "Main Menu"

In the Logiweb system each instance of a page has a so-called *main menu*, e.g., the main menu of this page is at:

`http://www.diku.dk/~grue/logiweb/20050502/home/mrmr/peano-commutativity/latest/`

From the main menu one can browse and find numerous kinds of information about the page. The most important of which to begin with is the page in `pdf` format and (for Logiweb beginners) the source code.

A Logiweb page is written in a mixture of LaTeX and the language `pyk` developed by Klaus Grue. `pyk` is the language in which the mathematics of a Logiweb page is written. The best way to get a feeling for this mixture is to read the source code of a Logiweb page. This is found under `Source` $\rightarrow$ `Actual Source` from the Logiweb main menu of a Logiweb page, e.g., the check page.

One can also read the `pyk` text of a page, which is found under `Body` $\rightarrow$ `Pyk`, also at the main menu. It is also under `Body` that one finds the TeX and pdf files of the page.[3]

[3]The TeX files are not available through a browser due to safety precautions.

3.2 The Pyk Language

As mentioned in the previous section, the `pyk` language has been developed for easy pronounciation or in general for "spoken mathematics", cf. the base page, Section 1.1. Thus, a mathematical expression we are well-acquaintated with, such as:

$$(x + y)^2 = x^2 + 2xy + y^2$$

may correspond to the following `pyk` expression:

> parenthesis var x plus var y end parenthesis square equals var x square plus two times var x times var y plus var y square[4]

Thus, eventually one will be able to "tell" one's mathematics to a computer and it will generate a file written in `pyk`, which will then be the source for precisely one Logiweb page. This is also the reason why `pyk` "is" case sensitive, that is, the `pyk` code cannot be in capital letters.

We thus see that `pyk` is much like any other programming language. However, in `pyk` one writes everything out in words instead of using symbols, e.g., "parenthesis" and "end parenthesis" instead of "(" and ")". In pdf format the textual representations will be replaced by their symbolic representations defined by the author of a Logiweb page. We shall return to this *aspect* of Logiweb in Section 4.2.

3.3 The Structure of a Logiweb Page

A Logiweb page has to be written according to certain standards. Thus, a specific structural sketch for a Logiweb page has to be used, which defines the specific and necessary parts of a Logiweb source file, that is, a `pyk` file. A `pyk` file has exactly one of each of these parts or sections, which will be described below. We note that the section names must be in capital letters. Like the TeX code and normal text written this is not part of `pyk` and does not need to be in lowercase letters.

`PAGE` A Logiweb page has a name which is given in the `PAGE` section of the `pyk` file. For instance, in the source code for this page one will find the PAGE section at the very top where it is seen that this page is named "peano commutativity". This also appears in Appendix A of this page where the acutal definition takes place.

`BIBLIOGRAPHY` The references of a Logiweb page are given in the `BIBLIOGRAPHY` section of the `pyk` file. A page can refer to every possible Logiweb page, also if one of these pages refers to once referenced pages. The Logiweb server will make sure no cyclic references occur. We note that the `BIBLIOGRAPHY` section of a Logiweb page is not the bibliography seen in Appendix **??**, which is the normal list of referenced books, articles etc.

[4] This example has been taken from [2].

`ASSOCIATIVITY` In the `ASSOCIATIVITY` section of the `pyk` file the constructs of a Logiweb page and their associativity—either POST- or PRE- are the options—are declared. The order of the constructs furthermore define their priority. Constructs declared with comma separation after the same associativity statement have the same priority. (In a sense the *constructs* of a Logiweb page make up the available `pyk` code in a page and/or pages referring to the page declaring and defining the constructs.) Thus, we get that:

```
PREASSOCIATIVE * plus *, * minus *
```

indicates that, e.g., [$\mathsf{x} + \mathsf{y}$] is to have the same priority as [$\mathsf{x} - \mathsf{y}$], whereas:

```
PREASSOCIATIVE * times *
⋮
PREASSOCIATIVE * plus *
```

indicates that, e.g., [$\mathsf{x} \cdot \mathsf{y}$] is to have higher priority than [$\mathsf{x} + \mathsf{y}$]. These examples are from the base page. In order to use these constructs in another page they should be entered in the associativity section of the referring page with `base` written after the associativity assignment. When using the names of constructs from a page in the associativity section of another page one is to place the priority of new constructs in proper hierarchical order. Thus, in the associativity section of the present page we put names of, e.g., the lemmas to be proved together with the `bracket` declaration, which gives those constructs the same priority:

```
PREASSOCIATIVE base:  bracket * end bracket,
mendelson lemma one eight eight
```

It might be confusing that `mendelson lemma one eight` and other constructs declared and defined on a certain page does not have the name of the page in front of the declaration in the associativity section, but one gets used to this and quickly learn to distinguish.

The associativites can be found in the priority table of Appendix B.

`BODY` In the `BODY` section we enter the actual text and math of the page. Hence, this is the section most readers get acquainted with, and they probably do so in pdf format. It is also in this section that the actual `pyk` coding is done.

We thus get the structure for a Logiweb page as is seen in Figure 1

The above only gives a bland taste of the richness of the very complex and impressive system that Logiweb is. But as stated earlier we believe it should be enough to get the reader started. To set up a system one is advised to read the `Pyk how-to` at:

```
PAGE name
⋮
BIBLIOGRAPHY
base: nani
          . . .
⋮
PREASSOCIATIVITY
⋮
BODY
⋮
```

Figure 1: The structure of a Logiweb page.

http://diku.dk/undervisning/2005s/235/howto.html

Now we proceed to the actual mathematics of this page, namely peano arithmetic. But before getting into details with this, we place the assignment in a larger frame of reference.

4 Axiomatic Theories

In this section we give a basic introduction to the concepts with which we will be working in the rest of the report. In may well be skipped by the (experienced) logician.

For simple systems of logic we can decide the truth value of a statement by mere use of truth tables. Rather quickly this turns out not to be enough when expressing more complex systems of logic. Though it took our ancestors a couple of millennia to give us the formalistic tools to express even the simplest of logical statements, the basic elements of the so-called formal axiomatic theories are quite simple. Cf. [4], pp. 34ff.

The branch of logic known as *propositional calculus* is simple to understand and serves well when introducing the concept of axiomatic theories. The abstract concept of propositions, i.e. the abstract denotation of sentences like "Jones is a communist" and "Jones is an atheist", and a few logical connectives, e.g. $\neg$ and $\Rightarrow$, can be combined with one of more rules of inference in a proper way and thus form a calculus. The combined logical statements can express many types of statements, the truth of which we can determine by use of truth tables. We extend these concepts in the following manner. We define a formal theory $\mathcal{L}$ when the following conditions are satisfied:[5]

1. In $\mathcal{L}$ a countable set of symbols are given, a finite sequence of which is called an *expression* of $\mathcal{L}$.

[5]We only state the most necessary concepts here, cf. [4], pp. 34f. for an elaborate explanation.

2. A subset of the set of expressions of $\mathcal{L}$ are called the set of *well-formed formulas* of $\mathcal{L}$ or *wfs* for short when written in plural.

3. A set of the wfs are called the *axioms* of $\mathcal{L}$. If it can effectively be decided whether a wf is an axiom then $\mathcal{L}$ is said to be an *axiomatic* theory.[6]

4. A finite set of relations, $\mathrm{R_1, \ldots, R_n}$, among the wfs make up the *rules of inference*. We say that the wf $\mathcal{B}$ is a *direct consequence* of a given set of wfs and that it is so by virtue of $\mathrm{R_i}$, if for each $\mathrm{R_i}$, there is a unique positive integer j such that, for every set of j wfs and each $\mathcal{B}$, one can effectively decide whether the given j wfs are in the relation $\mathrm{R_i}$ to $\mathcal{B}$.

A *proof* in $\mathcal{L}$ is a sequence $\mathcal{B}_1, \ldots, \mathcal{B}_\mathrm{k}$ of wfs such that, for each i, either $\mathcal{B}_\mathrm{i}$ is an axiom of $\mathcal{L}$ or $\mathcal{B}_\mathrm{i}$ is a direct consequence of some of the preceding wfs in the sequence by virtue of one of the rules of inference of $\mathcal{L}$.

A *theorem* of $\mathcal{L}$ is a wf $\mathcal{B}$ such that $\mathcal{B}$ is the last wf of some proof in $\mathcal{L}$. Such a proof is called a *proof of $\mathcal{B}$ in $\mathcal{L}$*. If it can be machine checked whether a given theorem is indeed a theorem of a given theory then we say that the theory is *decidable*, otherwise it is said to be *undecidable*.

We say that $\mathcal{C}$ is a *consequence* of a set of wfs, Γ, if and only if there is a finite sequence of wfs, $\mathcal{B}_1, \ldots, \mathcal{B}_\mathrm{k}$, such that $\mathcal{C}$ is $\mathcal{B}_\mathrm{k}$, and, for each i, either $\mathcal{B}_\mathrm{i}$ is an axiom or $\mathcal{B}_\mathrm{i}$ is a direct consequence by some rule of inference of some of the preceding wfs in the sequence, which is then called a *proof* or (*deduction*) of $\mathcal{C}$ from Γ. The members of Γ are called the *hypotheses* or *premisses* of the proof.

We now have the prerequisites in order to define an actual theory.

4.1 The Axiomatic Theory L

The concept of a formal axiomatic theory L for the propositional calculus can now be introduced:

1. The symbols of L are $\neg$, $\Rightarrow$, (,), and the letters $\mathrm{A_i}$, for $\mathrm{i} \in \mathrm{N}$.

2. Well-formed formulas:

   (a) All statement letters are wfs.

   (b) if $\mathcal{B}$ and $\mathcal{C}$ are wfs, then so are $(\neg\mathcal{B})$ and $(\mathcal{B} \Rightarrow \mathcal{C})$.

3. If $\mathcal{B}$, $\mathcal{C}$, and $\mathcal{D}$ are wfs of L, then the following are axioms of L:

   (A1) $(\mathcal{B} \Rightarrow (\mathcal{C} \Rightarrow \mathcal{B}))$

   (A2) $((\mathcal{B} \Rightarrow (\mathcal{C} \Rightarrow \mathcal{D})) \Rightarrow ((\mathcal{B} \Rightarrow \mathcal{C}) \Rightarrow (\mathcal{B} \Rightarrow \mathcal{D})))$

   (A3) $(((\neg\mathcal{C}) \Rightarrow (\neg\mathcal{B})) \Rightarrow (((\neg\mathcal{C}) \Rightarrow \mathcal{B}) \Rightarrow \mathcal{C}))$

4. One rule of inference of L, namely *modus ponens*, which says that $\mathcal{C}$ is a direct consequence of $\mathcal{B}$ and $\mathcal{B} \Rightarrow \mathcal{C}$. This rule is abbreviated as MP.

[6] We thus have that theories defined in Logiweb must be axiomatic theories.

This theory can, of course, be elaborated upon, but we shall not pursue the details of this any further. Nor shall we make more distinctions in the logical concepts with which this assignment is concerned. Instead we shall place it in the Logiweb frame of reference.

4.2 An Axiomatic Logiweb Theory

The conceptual framework introduced in the previous subsection is the one we shall work in when proving the law of commutativity of addition within peano arithmetic. Since this proof must be carried out within the system of Logiweb we here give a Logiweb example and prove a very basic lemma from [4], p. 36, namely [M Lemma 1.8][7]. (We shall adopt the convention of writing "M" in front of a lemma, proposition etc. to denote "Mendelson".)

Now, the appearence of "M Lemma 1.8" above is somewhat special. First, the name of the lemma (actually, it is not a lemma until defined to be so—now it is merely a *construct* in this page) appeared in brackets and with a footnote. This is specific to Logiweb and from it we are introduced to some of the *aspects* of Logiweb. Cf. the base page, Section 2.2.1.

If one reads the `pyk` file, one can find the declaration of [M Lemma 1.8] as being part of this Logiweb page almost at the very top, namely in the associativity section. And at this point in the text we introduce the construct to Logiweb, that is, from now on one can refer to the construct by writing some specific `pyk` code. One refers to this construct by writing what appears in the footnote of the construct after the equality sign with "pyk" written above it; this is an example of the *pyk aspect* of Logiweb. As is clear now, it simply defines the `pyk` code.

Another aspect is the *TEX aspect* of Logiweb, which is simply how a construct in Logiweb appears in writing, that is, in those formats one can build from a normal TEX document, e.g., a pdf document. The TEX aspect is defined by the user or author who within the limits of TEX single-handedly decides how the construct should appear.

Having made the construct visible or available to Logiweb, we want to place it in some specific framework within which we work, that is, in a theory or system. We could denote this system L in accordance with what has been said above and build our theory from the bottom. Instead we shall make use of the work of Klaus Grue. Thus, we introduce [M Lemma 1.8] in [S′], which is defined on the peano page. We do this as follows:

[S′ **lemma** M Lemma 1.8: $\forall \mathcal{B} \colon \mathcal{B} \dot{\Rightarrow} \mathcal{B}$]

For now we merely mention in passing that the special symbol $\dot{\Rightarrow}$ with the little dot appearing above the imply symbol tells us that this implication is specific to the peano arithmetic defined in [S′]. This system will be the one

[7] [M Lemma 1.8 $\overset{\text{pyk}}{=}$ "mendelson lemma one eight"]

within which we will be working, and we shall examine it more in the next section.

We have now defined the construct [M Lemma 1.8] to be a lemma in the system [S′]. Then we must prove it in order to use it later should we have the need for that. This is done using axiomatic constructs from the peano page. For now, we shall use these without further introduction, which we defer until the next section:

S′ **proof of** M Lemma 1.8:

L01:	Arbitrary $\gg$	$\mathcal{B}$	;
L02:	A2′ $\gg$	$(\mathcal{B} \dot{\Rightarrow} (\mathcal{B} \dot{\Rightarrow} \mathcal{B}) \dot{\Rightarrow} \mathcal{B}) \dot{\Rightarrow} ((\mathcal{B} \dot{\Rightarrow} (\mathcal{B} \dot{\Rightarrow} \mathcal{B})) \dot{\Rightarrow} (\mathcal{B} \dot{\Rightarrow} \mathcal{B}))$	;
L03:	A1′ $\gg$	$\mathcal{B} \dot{\Rightarrow} ((\mathcal{B} \dot{\Rightarrow} \mathcal{B}) \dot{\Rightarrow} \mathcal{B})$	;
L04:	MP′ $\rhd$ L02 $\rhd$ L03 $\gg$	$(\mathcal{B} \dot{\Rightarrow} (\mathcal{B} \dot{\Rightarrow} \mathcal{B})) \dot{\Rightarrow} (\mathcal{B} \dot{\Rightarrow} \mathcal{B})$	;
L05:	A1′ $\gg$	$\mathcal{B} \dot{\Rightarrow} (\mathcal{B} \dot{\Rightarrow} \mathcal{B})$	;
L06:	MP′ $\rhd$ L04 $\rhd$ L05 $\gg$	$\mathcal{B} \dot{\Rightarrow} \mathcal{B}$	$\square$

We have now seen an example of a proof of a lemma belonging to a certain theory in Logiweb. In the next section we will take a closer look at the axiomatic theory [S′] in which we will be working.

5 Peano Arithmetic

As stated above, the assignment of this report is to prove the law of commutativity of addition within the natural numbers. More precisely, we shall prove it within the system of *peano arithmetic* as it is stated in Chapter 3 of [4]. Peano arithmetic is based upon five postulates:

(P1) 0 is a natural number.

(P2) If x is a natural number, then there is another natural number denoted by x′; this is denoted the *successor* of x.)

(P3) $0 \neq$ x′ for every natural number x.

(P4) If x′ = y′, then x = y.

(P5) If Q is a property that may or may not hold for any given natural number, and if:

- (I) 0 has the property Q, and
- (II) whenever a natural number x has the property Q, then x′ has the property Q,

then all natural numbers have the property Q. This is the mathematical principle of induction.

Klaus Grue has implemented peano arithmetic making most of the needed constructs to solve the assignment available to us. Cf. peano, Section 1.5. The result of his work is among other things the theory within which we will be working, namely the aforementioned theory [S′]. This theory is much like the axiomatic theory we met in Section 4.1, p. 10. Only, this theory has more axioms, a number of theorems we can use, and one more rule of inference. We repeat the full system in the following for easy reference.

5.1 Peano Arithmetic in Logiweb

Peano arithmetic in Logiweb consists of the axioms, theorems, and rules of inference that are reproduced in the following subsections; they are introduced and defined in [3]. The appearance of the constructs is somewhat special. As mentioned above, some symbols are furnished with a little dot. This special TEX aspect need not confuse us. It only shows that we are working within peano arithmetic as defined in the peano page. Also, the equality symbol of peano arithmetic is furnished with a small p, which serves the same purpose as the dot.

5.1.1 [S′] Axioms

In peano arithmetic—or in [**Theory** S′] or merely [S′] as we shall refer to it from now on—the following five axioms or rules are available. We assume that the special symbols used are known to the reader.

[S′ **rule** A1′: $\forall \mathcal{A}\colon \forall \mathcal{B}\colon \mathcal{A} \mathrel{\dot{\Rightarrow}} \mathcal{B} \mathrel{\dot{\Rightarrow}} \mathcal{A}$]

[S′ **rule** A2′: $\forall \mathcal{A}\colon \forall \mathcal{B}\colon \forall \mathcal{C}\colon (\mathcal{A} \mathrel{\dot{\Rightarrow}} \mathcal{B} \mathrel{\dot{\Rightarrow}} \mathcal{C}) \mathrel{\dot{\Rightarrow}} (\mathcal{A} \mathrel{\dot{\Rightarrow}} \mathcal{B}) \mathrel{\dot{\Rightarrow}} \mathcal{A} \mathrel{\dot{\Rightarrow}} \mathcal{C}$]

[S′ **rule** A3′: $\forall \mathcal{A}\colon \forall \mathcal{B}\colon (\dot{\neg}\, \mathcal{B} \mathrel{\dot{\Rightarrow}} \dot{\neg}\, \mathcal{A}) \mathrel{\dot{\Rightarrow}} (\dot{\neg}\, \mathcal{B} \mathrel{\dot{\Rightarrow}} \mathcal{A}) \mathrel{\dot{\Rightarrow}} \mathcal{B}$]

[S′ **rule** A4′: $\forall \mathcal{C}\colon \forall \mathcal{A}\colon \forall \mathcal{X}\colon \forall \mathcal{B}\colon \lceil \mathcal{A} \rceil \equiv \langle \lceil \mathcal{B} \rceil | \lceil \mathcal{X} \rceil := \lceil \mathcal{C} \rceil \rangle \Vdash \dot{\forall} \mathcal{X}\colon \mathcal{B} \mathrel{\dot{\Rightarrow}} \mathcal{A}$]

[S′ **rule** A5′: $\forall \mathcal{X}\colon \forall \mathcal{A}\colon \forall \mathcal{B}\colon$
$\mathrm{nonfree}(\lceil \mathcal{X} \rceil, \lceil \mathcal{A} \rceil) \Vdash \dot{\forall} \mathcal{X}\colon (\mathcal{A} \mathrel{\dot{\Rightarrow}} \mathcal{B}) \mathrel{\dot{\Rightarrow}} \mathcal{A} \mathrel{\dot{\Rightarrow}} \dot{\forall} \mathcal{X}\colon \mathcal{B}$]

These definitions are copied from the peano page, Section 1.5.

5.1.2 [S′] Rules of Inference

Besides the five axioms [S′] has two rules of inference. The first is the well-known modus ponens, which all computer scientists are well-acquainted with. The second says that a well-formed formula has as a consequence that it is sound or valid for all variables, $\mathrm{x_i}$.

[S′ **rule** MP′: $\forall \mathcal{A}\colon \forall \mathcal{B}\colon \mathcal{A} \mathrel{\dot{\Rightarrow}} \mathcal{B} \vdash \mathcal{A} \vdash \mathcal{B}$]

[S′ **rule** Gen′: $\forall \mathcal{X}\colon \forall \mathcal{A}\colon \mathcal{A} \vdash \dot{\forall} \mathcal{X}\colon \mathcal{A}$]

These definitions are copied from the peano page, Section 1.5.

5.1.3 [S′] Theorems

Finally, within [S′] we are allowed to use the following theorems. We shall not prove these:

[S′ **rule** S1′: $\forall\mathcal{A}\colon\forall\mathcal{B}\colon\forall\mathcal{C}\colon \mathcal{A}\overset{\mathrm{p}}{=}\mathcal{B}\mathrel{\dot\Rightarrow}\mathcal{A}\overset{\mathrm{p}}{=}\mathcal{C}\mathrel{\dot\Rightarrow}\mathcal{B}\overset{\mathrm{p}}{=}\mathcal{C}$]

[S′ **rule** S2′: $\forall\mathcal{A}\colon\forall\mathcal{B}\colon \mathcal{A}\overset{\mathrm{p}}{=}\mathcal{B}\mathrel{\dot\Rightarrow}\mathcal{A}'\overset{\mathrm{p}}{=}\mathcal{B}'$]

[S′ **rule** S3′: $\forall\mathcal{A}\colon \dot\neg\dot{0}\overset{\mathrm{p}}{=}\mathcal{A}'$]

[S′ **rule** S4′: $\forall\mathcal{A}\colon\forall\mathcal{B}\colon \mathcal{A}'\overset{\mathrm{p}}{=}\mathcal{B}'\mathrel{\dot\Rightarrow}\mathcal{A}\overset{\mathrm{p}}{=}\mathcal{B}$]

[S′ **rule** S5′: $\forall\mathcal{A}\colon \mathcal{A}\mathbin{\dot+}\dot{0}\overset{\mathrm{p}}{=}\mathcal{A}$]

[S′ **rule** S6′: $\forall\mathcal{A}\colon\forall\mathcal{B}\colon \mathcal{A}\mathbin{\dot+}\mathcal{B}'\overset{\mathrm{p}}{=}(\mathcal{A}\mathbin{\dot+}\mathcal{B})'$]

[S′ **rule** S7′: $\forall\mathcal{A}\colon \mathcal{A}\mathbin{\dot\cdot}\dot{0}\overset{\mathrm{p}}{=}\dot{0}$]

[S′ **rule** S8′: $\forall\mathcal{A}\colon\forall\mathcal{B}\colon \mathcal{A}\mathbin{\dot\cdot}(\mathcal{B}')\overset{\mathrm{p}}{=}(\mathcal{A}\mathbin{\dot\cdot}\mathcal{B})\mathbin{\dot+}\mathcal{A}$]

[S′ **rule** S9′: $\forall\mathcal{A}\colon\forall\mathcal{B}\colon\forall\mathcal{C}\colon\forall\mathcal{X}\colon$
$\mathcal{B}\equiv\langle\mathcal{A}|\mathcal{X}:=\dot{0}\rangle\Vdash\mathcal{C}\equiv\langle\mathcal{A}|\mathcal{X}:=\mathcal{X}'\rangle\Vdash$
$\mathcal{B}\mathrel{\dot\Rightarrow}\dot\forall\mathcal{X}\colon(\mathcal{A}\mathrel{\dot\Rightarrow}\mathcal{C})\mathrel{\dot\Rightarrow}\dot\forall\mathcal{X}\colon\mathcal{A}$]

These definitions are copied from the peano page, Section 1.5.

We now have the necessary means to solve our assignment, which we will do in the following sections.

6 Proof of Commutativity of Addition

As mentioned in the introduction we shall prove the law of commutativity of addition in as simple a way as possible. We thus follow Mendelson in his proof, cf. [4], pp. 156ff. Hence, we will prove this law by proving Proposition 3.2(a)-(h), except Proposition 3.2(e). As we close in on our final goal we will be needing two tautologies and in our last proof a lemma, which we will state and prove as we go along. We do this in the following subsections. We explain what goes on as we go along. In each proof we first introduce the proposition in Logiweb, then we state the proposition, and finally we prove it.

6.1 M Proposition 3.2(a)

The first proposition states that a wf is equal to itself. Since peano arithmetic is theory with equality this is a basic and expected proposition—and quite logical!

[M Proposition 3.2(a)][8]

[S′ **lemma** M Proposition 3.2(a): $\forall \mathcal{T} \colon \mathcal{T} \overset{\mathrm{p}}{=} \mathcal{T}$]

S′ **proof of** M Proposition 3.2(a):

L01:	Arbitrary $\gg$	$\mathcal{T}$	;
L02:	S5′ $\gg$	$\mathcal{T} \dot{+} \dot{0} \overset{\mathrm{p}}{=} \mathcal{T}$	;
L03:	S1′ $\gg$	$(\mathcal{T} \dot{+} \dot{0} \overset{\mathrm{p}}{=} \mathcal{T}) \mathrel{\dot{\Rightarrow}} (\mathcal{T} \dot{+} \dot{0} \overset{\mathrm{p}}{=} \mathcal{T} \mathrel{\dot{\Rightarrow}} \mathcal{T} \overset{\mathrm{p}}{=} \mathcal{T})$	;
L04:	MP′ $\rhd$ L03 $\rhd$ L02 $\gg$	$\mathcal{T} \dot{+} \dot{0} \overset{\mathrm{p}}{=} \mathcal{T} \mathrel{\dot{\Rightarrow}} \mathcal{T} \overset{\mathrm{p}}{=} \mathcal{T}$	;
L05:	MP′ $\rhd$ L04 $\rhd$ L02 $\gg$	$\mathcal{T} \overset{\mathrm{p}}{=} \mathcal{T}$	$\square$

[Proposition 3.2(a)] is used in the proof of [Proposition 3.2(b)]. But before we can proceed to this we must prove a tautology also.

6.2 M Tautology A

In [4], p. 157 in the proof of [Proposition 3.2(b)] we see that a tautology is used. We shall state and prove this tautology for use in our next and later proofs.

[M Tautology A][9]

[S′ **lemma** M Tautology A: $\forall \mathcal{A} \colon \forall \mathcal{B} \colon \forall \mathcal{C} \colon \mathcal{A} \mathrel{\dot{\Rightarrow}} (\mathcal{B} \mathrel{\dot{\Rightarrow}} \mathcal{C}) \vdash \mathcal{B} \mathrel{\dot{\Rightarrow}} (\mathcal{A} \mathrel{\dot{\Rightarrow}} \mathcal{C})$]

S′ **proof of** M Tautology A:

L01:	Arbitrary $\gg$	$\mathcal{A}$	;
L02:	Arbitrary $\gg$	$\mathcal{B}$	;
L03:	Arbitrary $\gg$	$\mathcal{C}$	;
L04:	Premise $\gg$	$\mathcal{A} \mathrel{\dot{\Rightarrow}} (\mathcal{B} \mathrel{\dot{\Rightarrow}} \mathcal{C})$	;
L05:	A2′ $\gg$	$(\mathcal{A} \mathrel{\dot{\Rightarrow}} (\mathcal{B} \mathrel{\dot{\Rightarrow}} \mathcal{C})) \mathrel{\dot{\Rightarrow}} ((\mathcal{A} \mathrel{\dot{\Rightarrow}} \mathcal{B}) \mathrel{\dot{\Rightarrow}} (\mathcal{A} \mathrel{\dot{\Rightarrow}} \mathcal{C}))$	;
L06:	MP′ $\rhd$ L05 $\rhd$ L04 $\gg$	$(\mathcal{A} \mathrel{\dot{\Rightarrow}} \mathcal{B}) \mathrel{\dot{\Rightarrow}} (\mathcal{A} \mathrel{\dot{\Rightarrow}} \mathcal{C})$	;
L07:	A1′ $\gg$	$((\mathcal{A} \mathrel{\dot{\Rightarrow}} \mathcal{B}) \mathrel{\dot{\Rightarrow}} (\mathcal{A} \mathrel{\dot{\Rightarrow}} \mathcal{C})) \mathrel{\dot{\Rightarrow}} ((\mathcal{B} \mathrel{\dot{\Rightarrow}} (\mathcal{A} \mathrel{\dot{\Rightarrow}} \mathcal{B}) \mathrel{\dot{\Rightarrow}} (\mathcal{A} \mathrel{\dot{\Rightarrow}} \mathcal{C})))$	;
L08:	MP′ $\rhd$ L07 $\rhd$ L06 $\gg$	$\mathcal{B} \mathrel{\dot{\Rightarrow}} (\mathcal{A} \mathrel{\dot{\Rightarrow}} \mathcal{B}) \mathrel{\dot{\Rightarrow}} (\mathcal{A} \mathrel{\dot{\Rightarrow}} \mathcal{C})$	;
L09:	A2′ $\gg$	$(\mathcal{B} \mathrel{\dot{\Rightarrow}} (\mathcal{A} \mathrel{\dot{\Rightarrow}} \mathcal{B}) \mathrel{\dot{\Rightarrow}} (\mathcal{A} \mathrel{\dot{\Rightarrow}} \mathcal{C})) \mathrel{\dot{\Rightarrow}} (\mathcal{B} \mathrel{\dot{\Rightarrow}} (\mathcal{A} \mathrel{\dot{\Rightarrow}} \mathcal{B})) \mathrel{\dot{\Rightarrow}} (\mathcal{B} \mathrel{\dot{\Rightarrow}} (\mathcal{A} \mathrel{\dot{\Rightarrow}} \mathcal{C}))$	;
L10:	MP′ $\rhd$ L09 $\rhd$ L08 $\gg$	$(\mathcal{B} \mathrel{\dot{\Rightarrow}} (\mathcal{A} \mathrel{\dot{\Rightarrow}} \mathcal{B})) \mathrel{\dot{\Rightarrow}} (\mathcal{B} \mathrel{\dot{\Rightarrow}} (\mathcal{A} \mathrel{\dot{\Rightarrow}} \mathcal{C}))$	;
L11:	A1′ $\gg$	$\mathcal{B} \mathrel{\dot{\Rightarrow}} (\mathcal{A} \mathrel{\dot{\Rightarrow}} \mathcal{B})$	;
L12:	MP′ $\rhd$ L10 $\rhd$ L11 $\gg$	$\mathcal{B} \mathrel{\dot{\Rightarrow}} (\mathcal{A} \mathrel{\dot{\Rightarrow}} \mathcal{C})$	$\square$

[8] [M Proposition 3.2(a) $\overset{\text{pyk}}{=}$ “mendelson proposition three two a”]

[9] [M Tautology A $\overset{\text{pyk}}{=}$ “mendelson tautology a”]

We note that [M Proposition 3.2(b)] could also be proved by means of Corollary 1.10(b), cf. [4], p. 38. Mendelson proves this by the use of Proposition 1.9 also known as the Deduction Theorem, cf. [4], p. 37. We avoid the use of this, though, it would have been prettier to follow Mendelson completely.

6.3 M Proposition 3.2(b)

We now proceed to proving [M Proposition 3.2(b)] where we will make use of [M Tautology A].

[M Proposition 3.2(b)][10]

[S′ **lemma** M Proposition 3.2(b): $\forall \mathcal{T} \colon \forall \mathcal{R} \colon \mathcal{T} \overset{\text{p}}{=} \mathcal{R} \dot{\Rightarrow} \mathcal{R} \overset{\text{p}}{=} \mathcal{T}$]

S′ **proof of** M Proposition 3.2(b):

L01:	Arbitrary $\gg$	$\mathcal{T}$	;
L02:	Arbitrary $\gg$	$\mathcal{R}$	;
L03:	S1′ $\gg$	$\mathcal{T} \overset{\text{p}}{=} \mathcal{R} \dot{\Rightarrow} (\mathcal{T} \overset{\text{p}}{=} \mathcal{T} \dot{\Rightarrow} \mathcal{R} \overset{\text{p}}{=} \mathcal{T})$	;
L04:	M Tautology A $\rhd$ L03 $\gg$	$\mathcal{T} \overset{\text{p}}{=} \mathcal{T} \dot{\Rightarrow} (\mathcal{T} \overset{\text{p}}{=} \mathcal{R} \dot{\Rightarrow} \mathcal{R} \overset{\text{p}}{=} \mathcal{T})$	;
L05:	M Proposition 3.2(a) $\gg$	$\mathcal{T} \overset{\text{p}}{=} \mathcal{T}$	;
L06:	MP′ $\rhd$ L04 $\rhd$ L05 $\gg$	$\mathcal{T} \overset{\text{p}}{=} \mathcal{R} \dot{\Rightarrow} \mathcal{R} \overset{\text{p}}{=} \mathcal{T}$	$\square$

Thus having proved [M Proposition 3.2(b)] we proceed to [Proposition 3.2(c)] in the proof of which the former is used. But before we do this we need yet another tautology.

6.4 M Tautology B

In [4], p. 157 in the proof of M Proposition 3.2(c) we see that a tautology is used. We shall state and prove this tautology for use in our next and later proofs.

[M Tautology B][11]

[S′ **lemma** M Tautology B: $\forall \mathcal{D} \colon \forall \mathcal{E} \colon \forall \mathcal{F} \colon \mathcal{D} \dot{\Rightarrow} \mathcal{E} \vdash \mathcal{E} \dot{\Rightarrow} \mathcal{F} \vdash \mathcal{D} \dot{\Rightarrow} \mathcal{F}$]

S′ **proof of** M Tautology B:

L01:	Arbitrary $\gg$	$\mathcal{D}$	;
L02:	Arbitrary $\gg$	$\mathcal{E}$	;
L03:	Arbitrary $\gg$	$\mathcal{F}$	;
L04:	Premise $\gg$	$\mathcal{D} \dot{\Rightarrow} \mathcal{E}$	;
L05:	Premise $\gg$	$\mathcal{E} \dot{\Rightarrow} \mathcal{F}$	;
L06:	A1′ $\gg$	$(\mathcal{E} \dot{\Rightarrow} \mathcal{F}) \dot{\Rightarrow} (\mathcal{D} \dot{\Rightarrow} (\mathcal{E} \dot{\Rightarrow} \mathcal{F}))$	;

[10] [M Proposition 3.2(b) $\overset{\text{pyk}}{=}$ “mendelson proposition three two b”]

[11] [M Tautology B $\overset{\text{pyk}}{=}$ “mendelson tautology b”]

L07:	$\mathrm{MP}' \rhd \mathrm{L06} \rhd \mathrm{L05} \gg$	$\mathcal{D} \dot{\Rightarrow} (\mathcal{E} \dot{\Rightarrow} \mathcal{F})$	;
L08:	$\mathrm{A2}' \gg$	$(\mathcal{D} \dot{\Rightarrow} (\mathcal{E} \dot{\Rightarrow} \mathcal{F})) \dot{\Rightarrow} ((\mathcal{D} \dot{\Rightarrow} \mathcal{E}) \dot{\Rightarrow} (\mathcal{D} \dot{\Rightarrow} \mathcal{F}))$	;
L09:	$\mathrm{MP}' \rhd \mathrm{L08} \rhd \mathrm{L07} \gg$	$(\mathcal{D} \dot{\Rightarrow} \mathcal{E}) \dot{\Rightarrow} (\mathcal{D} \dot{\Rightarrow} \mathcal{F})$	;
L10:	$\mathrm{MP}' \rhd \mathrm{L09} \rhd \mathrm{L04} \gg$	$\mathcal{D} \dot{\Rightarrow} \mathcal{F}$	$\square$

We can now prove [M Proposition 3.2(c)].

6.5 M Proposition 3.2(c)

As noted above we will make use of [M Tautology B] in our proof of [Proposition 3.2(c)], which concerns transitivity.

[M Proposition 3.2(c)][12]

[S′ **lemma** M Proposition 3.2(c): $\forall \mathcal{T} \colon \forall \mathcal{R} \colon \forall \mathcal{S} \colon \mathcal{T} \overset{\mathrm{p}}{=} \mathcal{R} \dot{\Rightarrow} (\mathcal{R} \overset{\mathrm{p}}{=} \mathcal{S} \dot{\Rightarrow} \mathcal{T} \overset{\mathrm{p}}{=} \mathcal{S})$]

S′ **proof of** M Proposition 3.2(c):

L01:	Arbitrary $\gg$	$\mathcal{T}$	;
L02:	Arbitrary $\gg$	$\mathcal{R}$	;
L03:	Arbitrary $\gg$	$\mathcal{S}$	;
L04:	$\mathrm{S1}' \gg$	$\mathcal{R} \overset{\mathrm{p}}{=} \mathcal{T} \dot{\Rightarrow} (\mathcal{R} \overset{\mathrm{p}}{=} \mathcal{S} \dot{\Rightarrow} \mathcal{T} \overset{\mathrm{p}}{=} \mathcal{S})$	;
L05:	M Proposition 3.2(b) $\gg$	$\mathcal{T} \overset{\mathrm{p}}{=} \mathcal{R} \dot{\Rightarrow} \mathcal{R} \overset{\mathrm{p}}{=} \mathcal{T}$	;
L06:	M Tautology B $\rhd$ L05 $\rhd$ L04 $\gg$	$\mathcal{T} \overset{\mathrm{p}}{=} \mathcal{R} \dot{\Rightarrow} (\mathcal{R} \overset{\mathrm{p}}{=} \mathcal{S} \dot{\Rightarrow} \mathcal{T} \overset{\mathrm{p}}{=} \mathcal{S})$	$\square$

Thus having proved [M Proposition 3.2(c)] we now proceed with [Proposition 3.2(d)].

6.6 M Proposition 3.2(d)

In order to prove [M Proposition 3.2(d)] following the proof in [4], p. 157, we will be using two tautologies. A closer examination of the proofs reveals that the two tautologies are an instance of [M Tautology A] and one of [M Tautology B]. Furthermore, we will be needing [M Proposition 3.2(b)] in our proof as well. Hence, we can commence with the proof right away.

[M Proposition 3.2(d)][13]

[S′ **lemma** M Proposition 3.2(d): $\forall \mathcal{R} \colon \forall \mathcal{T} \colon \forall \mathcal{S} \colon \mathcal{R} \overset{\mathrm{p}}{=} \mathcal{T} \dot{\Rightarrow} (\mathcal{S} \overset{\mathrm{p}}{=} \mathcal{T} \dot{\Rightarrow} \mathcal{R} \overset{\mathrm{p}}{=} \mathcal{S})$]

S′ **proof of** M Proposition 3.2(d):

L01:	Arbitrary $\gg$	$\mathcal{R}$	;
L02:	Arbitrary $\gg$	$\mathcal{T}$	;

[12] [M Proposition 3.2(c) $\overset{\mathrm{pyk}}{=}$ "mendelson proposition three two c"]

[13] [M Proposition 3.2(d) $\overset{\mathrm{pyk}}{=}$ "mendelson proposition three two d"]

L03:	Arbitrary $\gg$	$\mathcal{S}$	;
L04:	M Proposition 3.2(c) $\gg$	$\mathcal{R} \overset{\mathrm{p}}{=} \mathcal{T} \dot{\Rightarrow} (\mathcal{T} \overset{\mathrm{p}}{=} \mathcal{S} \dot{\Rightarrow} \mathcal{R} \overset{\mathrm{p}}{=} \mathcal{S})$	;
L05:	M Tautology A $\rhd$ L04 $\gg$	$\mathcal{T} \overset{\mathrm{p}}{=} \mathcal{S} \dot{\Rightarrow} (\mathcal{R} \overset{\mathrm{p}}{=} \mathcal{T} \dot{\Rightarrow} \mathcal{R} \overset{\mathrm{p}}{=} \mathcal{S})$	;
L06:	M Proposition 3.2(b) $\gg$	$\mathcal{S} \overset{\mathrm{p}}{=} \mathcal{T} \dot{\Rightarrow} \mathcal{T} \overset{\mathrm{p}}{=} \mathcal{S}$	;
L07:	M Tautology B $\rhd$ L06 $\rhd$ L05 $\gg$	$\mathcal{S} \overset{\mathrm{p}}{=} \mathcal{T} \dot{\Rightarrow} (\mathcal{R} \overset{\mathrm{p}}{=} \mathcal{T} \dot{\Rightarrow} \mathcal{R} \overset{\mathrm{p}}{=} \mathcal{S})$	;
L08:	M Tautology A $\rhd$ L07 $\gg$	$\mathcal{R} \overset{\mathrm{p}}{=} \mathcal{T} \dot{\Rightarrow} (\mathcal{S} \overset{\mathrm{p}}{=} \mathcal{T} \dot{\Rightarrow} \mathcal{R} \overset{\mathrm{p}}{=} \mathcal{S})$	$\square$

Thus having proved [M Proposition 3.2(d)] we now proceed with [Proposition 3.2(f)] in the proof of which we use the result just obtained.

6.7 M Proposition 3.2(f)

The proof of [M Proposition 3.2(f)] is split up into two parts as is seen from [4], pp. 157f. Furthermore, we now make use of regular variables and not metavariables as was the case in the previous proofs. That is, we now show our theorem to be true not for all terms but for all variables. This does not make a difference, though.

6.7.1 M Proposition 3.2(f) (i)

After having introduced the proposition and placed it in the theory in which we are working, we first show the identity of [$\dot{0}$]. This is the base case of a well-known proof by induction. After have proved the base case, we shall thus proceed and prove the induction step by the use of our two tautologies. Finally, we prove [M Proposition 3.2(f)] by the use of our two intermediary results. This structure is used in the proofs of the rest of the theorems including the final, namely, [Proposition 3.2(h)] stating the commutative law of addition.

[M Proposition 3.2(f) (i)][14]

[S′ **lemma** M Proposition 3.2(f) (i): $\dot{0} \overset{\mathrm{p}}{=} \dot{0} \dot{+} \dot{0}$]

S′ **proof of** M Proposition 3.2(f) (i):

L01:	S5′ $\gg$	$\dot{0} \dot{+} \dot{0} \overset{\mathrm{p}}{=} \dot{0}$	;
L02:	M Proposition 3.2(b) $\gg$	$\dot{0} \dot{+} \dot{0} \overset{\mathrm{p}}{=} \dot{0} \dot{\Rightarrow} \dot{0} \overset{\mathrm{p}}{=} \dot{0} \dot{+} \dot{0}$	;
L03:	MP′ $\rhd$ L02 $\rhd$ L01 $\gg$	$\dot{0} \overset{\mathrm{p}}{=} \dot{0} \dot{+} \dot{0}$	$\square$

We have now showed the base case in an induction. The next step is to continue the induction.

6.7.2 M Proposition 3.2(f) (ii)

In Mendelson's proof of [M Proposition 3.2(f) (ii)] we note that the Deduction Theorem is used, cf. [4], p. 158. We shall not use this theorem in our proof but

[14] [M Proposition 3.2(f) (i) $\overset{\mathrm{pyk}}{=}$ "mendelson proposition three two f i"]

instead use the previously proved tautologies.

[M Proposition 3.2(f) (ii)][15]

[S′ **lemma** M Proposition 3.2(f) (ii): $\dot{\forall}\dot{x}\colon (\dot{x} \stackrel{p}{=} \dot{0} \dot{+} \dot{x} \dot{\Rightarrow} \dot{x}' \stackrel{p}{=} \dot{0} \dot{+} \dot{x}')$]

S′ **proof of** M Proposition 3.2(f) (ii):

L01:	S6′ $\gg$	$\dot{0} \dot{+} \dot{x}' \stackrel{p}{=} (\dot{0} \dot{+} \dot{x})'$	;
L02:	S2′ $\gg$	$\dot{x} \stackrel{p}{=} \dot{0} \dot{+} \dot{x} \dot{\Rightarrow} \dot{x}' \stackrel{p}{=} (\dot{0} \dot{+} \dot{x})'$	;
L03:	M Proposition 3.2(d) $\gg$	$\dot{x}' \stackrel{p}{=} (\dot{0} \dot{+} \dot{x})' \dot{\Rightarrow} (\dot{0} \dot{+} \dot{x}' \stackrel{p}{=} (\dot{0} \dot{+} \dot{x})' \dot{\Rightarrow} \dot{x}' \stackrel{p}{=} \dot{0} \dot{+} \dot{x}')$	;
L04:	M Tautology B $\rhd$ L02 $\rhd$ L03 $\gg$	$\dot{x} \stackrel{p}{=} \dot{0} \dot{+} \dot{x} \dot{\Rightarrow} (\dot{0} \dot{+} \dot{x}' \stackrel{p}{=} (\dot{0} \dot{+} \dot{x})' \dot{\Rightarrow} \dot{x}' \stackrel{p}{=} \dot{0} \dot{+} \dot{x}')$	;
L05:	M Tautology A $\rhd$ L04 $\gg$	$\dot{0} \dot{+} \dot{x}' \stackrel{p}{=} (\dot{0} \dot{+} \dot{x})' \dot{\Rightarrow} (\dot{x} \stackrel{p}{=} \dot{0} \dot{+} \dot{x} \dot{\Rightarrow} \dot{x}' \stackrel{p}{=} \dot{0} \dot{+} \dot{x}')$	;
L06:	MP′ $\rhd$ L05 $\rhd$ L01 $\gg$	$\dot{x} \stackrel{p}{=} \dot{0} \dot{+} \dot{x} \dot{\Rightarrow} \dot{x}' \stackrel{p}{=} \dot{0} \dot{+} \dot{x}'$	;
L07:	Gen′ $\rhd$ L06 $\gg$	$\dot{\forall}\dot{x}\colon (\dot{x} \stackrel{p}{=} \dot{0} \dot{+} \dot{x} \dot{\Rightarrow} \dot{x}' \stackrel{p}{=} \dot{0} \dot{+} \dot{x}')$	□

We now have the necessary foundation to prove [M Proposition 3.2(f)].

6.7.3 M Proposition 3.2(f)

As is seen from [4], p. 158, induction is used in the proof of [M Proposition 3.2(f)]. Due to our ambition of keeping things as simple as possible, we have not implemented an induction theorem. Hence, the proof will deviate from the one given by Mendelson.

[M Proposition 3.2(f)][16]

[S′ **lemma** M Proposition 3.2(f): $\dot{\forall}\dot{x}\colon (\dot{x} \stackrel{p}{=} \dot{0} \dot{+} \dot{x})$]

S′ **proof of** M Proposition 3.2(f):

L01:	S9′ $\gg$	$(\dot{0} \stackrel{p}{=} \dot{0} \dot{+} \dot{0}) \dot{\Rightarrow} \dot{\forall}\dot{x}\colon (\dot{x} \stackrel{p}{=} \dot{0} \dot{+} \dot{x} \dot{\Rightarrow} \dot{x}' \stackrel{p}{=} \dot{0} \dot{+} \dot{x}') \dot{\Rightarrow} \dot{\forall}\dot{x}\colon (\dot{x} \stackrel{p}{=} \dot{0} \dot{+} \dot{x})$	;
L02:	M Proposition 3.2(f) (i) $\gg$	$\dot{0} \stackrel{p}{=} \dot{0} \dot{+} \dot{0}$	;
L03:	MP′ $\rhd$ L01 $\rhd$ L02 $\gg$	$\dot{\forall}\dot{x}\colon (\dot{x} \stackrel{p}{=} \dot{0} \dot{+} \dot{x} \dot{\Rightarrow} \dot{x}' \stackrel{p}{=} \dot{0} \dot{+} \dot{x}') \dot{\Rightarrow} \dot{\forall}\dot{x}\colon (\dot{x} \stackrel{p}{=} \dot{0} \dot{+} \dot{x})$	;
L04:	M Proposition 3.2(f) (ii) $\gg$	$\dot{\forall}\dot{x}\colon (\dot{x} \stackrel{p}{=} \dot{0} \dot{+} \dot{x} \dot{\Rightarrow} \dot{x}' \stackrel{p}{=} \dot{0} \dot{+} \dot{x}')$	;
L05:	MP′ $\rhd$ L03 $\rhd$ L04 $\gg$	$\dot{\forall}\dot{x}\colon (\dot{x} \stackrel{p}{=} \dot{0} \dot{+} \dot{x})$	□

[15] [M Proposition 3.2(f) (ii) $\stackrel{\text{pyk}}{=}$ “mendelson proposition three two f ii”]

[16] [M Proposition 3.2(f) $\stackrel{\text{pyk}}{=}$ “mendelson proposition three two f”]

We now proceed to proving [M Proposition 3.2(g)], which is used in our final proof.

6.8 M Proposition 3.2(g)

The proof of [M Proposition 3.2(g)] is similar to our previous proof. In this case, too, we need a base case and an induction step.

6.8.1 M Proposition 3.2(g) (i)

After having introduced the proposition and placed it in the theory in which we are working, we first show the base case.

[M Proposition 3.2(g) (i)][17]

[S′ **lemma** M Proposition 3.2(g) (i): $\dot\forall\dot x\colon(\dot x' \dot+ \dot 0 \overset{\mathrm{p}}{=} (\dot x \dot+ \dot 0)')$]

S′ **proof of** M Proposition 3.2(g) (i):

L01: S5′ ≫ $\dot x' \dot+ \dot 0 \overset{\mathrm{p}}{=} \dot x'$;

L02: S5′ ≫ $\dot x \dot+ \dot 0 \overset{\mathrm{p}}{=} \dot x$;

L03: S2′ ≫ $\dot x \dot+ \dot 0 \overset{\mathrm{p}}{=} \dot x \dot\Rightarrow (\dot x \dot+ \dot 0)' \overset{\mathrm{p}}{=} \dot x'$;

L04: MP′ ▷ L03 ▷ L02 ≫ $(\dot x \dot+ \dot 0)' \overset{\mathrm{p}}{=} \dot x'$;

L05: M Proposition 3.2(d) ≫ $\dot x' \dot+ \dot 0 \overset{\mathrm{p}}{=} \dot x' \dot\Rightarrow (\dot x \dot+ \dot 0)' \overset{\mathrm{p}}{=} \dot x' \dot\Rightarrow \dot x' \dot+ \dot 0 \overset{\mathrm{p}}{=} (\dot x \dot+ \dot 0)'$;

L06: MP′ ▷ L05 ▷ L01 ≫ $(\dot x \dot+ \dot 0)' \overset{\mathrm{p}}{=} \dot x' \dot\Rightarrow \dot x' \dot+ \dot 0 \overset{\mathrm{p}}{=} (\dot x \dot+ \dot 0)'$;

L07: MP′ ▷ L06 ▷ L04 ≫ $\dot x' \dot+ \dot 0 \overset{\mathrm{p}}{=} (\dot x \dot+ \dot 0)'$;

L08: Gen′ ▷ L07 ≫ $\dot\forall\dot x\colon(\dot x' \dot+ \dot 0 \overset{\mathrm{p}}{=} (\dot x \dot+ \dot 0)')$ □

6.8.2 M Proposition 3.2(g) (ii)

We now proceed with the induction step.
[M Proposition 3.2(g) (ii)][18]

[S′ **lemma** M Proposition 3.2(g) (ii): $\dot\forall\dot y\colon(\dot x' \dot+ \dot y \overset{\mathrm{p}}{=} (\dot x \dot+ \dot y)' \dot\Rightarrow \dot x' \dot+ \dot y' \overset{\mathrm{p}}{=} (\dot x \dot+ \dot y')')$]

S′ **proof of** M Proposition 3.2(g) (ii):

L01: S6′ ≫ $\dot x' \dot+ \dot y' \overset{\mathrm{p}}{=} (\dot x' \dot+ \dot y)'$;

L02: S2′ ≫ $\dot x' \dot+ \dot y \overset{\mathrm{p}}{=} (\dot x \dot+ \dot y)' \dot\Rightarrow (\dot x' \dot+ \dot y)' \overset{\mathrm{p}}{=} (\dot x \dot+ \dot y)''$;

[17] [M Proposition 3.2(g) (i) $\overset{\mathrm{pyk}}{=}$ "mendelson proposition three two g i"]

[18] [M Proposition 3.2(g) (ii) $\overset{\mathrm{pyk}}{=}$ "mendelson proposition three two g ii"]

L03: M Proposition 3.2(c) $\gg$ $\dot{x}' \dot{+} \dot{y}' \overset{\mathrm{p}}{=} (\dot{x}' \dot{+} \dot{y})' \dot{\Rightarrow} (\dot{x}' \dot{+} \dot{y})' \overset{\mathrm{p}}{=} (\dot{x} \dot{+} \dot{y})'' \dot{\Rightarrow} \dot{x}' \dot{+} \dot{y}' \overset{\mathrm{p}}{=} (\dot{x} \dot{+} \dot{y})''$;

L04: M Tautology A $\rhd$ L03 $\gg$ $(\dot{x}' \dot{+} \dot{y})' \overset{\mathrm{p}}{=} (\dot{x} \dot{+} \dot{y})'' \dot{\Rightarrow} \dot{x}' \dot{+} \dot{y}' \overset{\mathrm{p}}{=} (\dot{x}' \dot{+} \dot{y})' \dot{\Rightarrow} \dot{x}' \dot{+} \dot{y}' \overset{\mathrm{p}}{=} (\dot{x} \dot{+} \dot{y})''$;

L05: M Tautology B $\rhd$ L02 $\rhd$ L04 $\gg$ $\dot{x}' \dot{+} \dot{y} \overset{\mathrm{p}}{=} (\dot{x} \dot{+} \dot{y})' \dot{\Rightarrow} \dot{x}' \dot{+} \dot{y}' \overset{\mathrm{p}}{=} (\dot{x}' \dot{+} \dot{y})' \dot{\Rightarrow} \dot{x}' \dot{+} \dot{y}' \overset{\mathrm{p}}{=} (\dot{x} \dot{+} \dot{y})''$;

L06: M Tautology A $\rhd$ L05 $\gg$ $\dot{x}' \dot{+} \dot{y}' \overset{\mathrm{p}}{=} (\dot{x}' \dot{+} \dot{y})' \dot{\Rightarrow} \dot{x}' \dot{+} \dot{y} \overset{\mathrm{p}}{=} (\dot{x} \dot{+} \dot{y})' \dot{\Rightarrow} \dot{x}' \dot{+} \dot{y}' \overset{\mathrm{p}}{=} (\dot{x} \dot{+} \dot{y})''$;

L07: MP$'$ $\rhd$ L06 $\rhd$ L01 $\gg$ $\dot{x}' \dot{+} \dot{y} \overset{\mathrm{p}}{=} (\dot{x} \dot{+} \dot{y})' \dot{\Rightarrow} \dot{x}' \dot{+} \dot{y}' \overset{\mathrm{p}}{=} (\dot{x} \dot{+} \dot{y})''$;

L08: S6$'$ $\gg$ $\dot{x} \dot{+} \dot{y}' \overset{\mathrm{p}}{=} (\dot{x} \dot{+} \dot{y})'$;

L09: S2$'$ $\gg$ $\dot{x} \dot{+} \dot{y}' \overset{\mathrm{p}}{=} (\dot{x} \dot{+} \dot{y})' \dot{\Rightarrow} (\dot{x} \dot{+} \dot{y}')' \overset{\mathrm{p}}{=} (\dot{x} \dot{+} \dot{y})''$;

L10: M Proposition 3.2(d) $\gg$ $\dot{x}' \dot{+} \dot{y}' \overset{\mathrm{p}}{=} (\dot{x} \dot{+} \dot{y})'' \dot{\Rightarrow} (\dot{x} \dot{+} \dot{y}')' \overset{\mathrm{p}}{=} (\dot{x} \dot{+} \dot{y})'' \dot{\Rightarrow} \dot{x}' \dot{+} \dot{y}' \overset{\mathrm{p}}{=} (\dot{x} \dot{+} \dot{y}')'$;

L11: M Tautology B $\rhd$ L07 $\rhd$ L10 $\gg$ $\dot{x}' \dot{+} \dot{y} \overset{\mathrm{p}}{=} (\dot{x} \dot{+} \dot{y})' \dot{\Rightarrow} (\dot{x} \dot{+} \dot{y}')' \overset{\mathrm{p}}{=} (\dot{x} \dot{+} \dot{y})'' \dot{\Rightarrow} \dot{x}' \dot{+} \dot{y}' \overset{\mathrm{p}}{=} (\dot{x} \dot{+} \dot{y}')'$;

L12: M Tautology A $\rhd$ L11 $\gg$ $(\dot{x} \dot{+} \dot{y}')' \overset{\mathrm{p}}{=} (\dot{x} \dot{+} \dot{y})'' \dot{\Rightarrow} \dot{x}' \dot{+} \dot{y} \overset{\mathrm{p}}{=} (\dot{x} \dot{+} \dot{y})' \dot{\Rightarrow} \dot{x}' \dot{+} \dot{y}' \overset{\mathrm{p}}{=} (\dot{x} \dot{+} \dot{y}')'$;

L13: MP$'$ $\rhd$ L09 $\rhd$ L08 $\gg$ $(\dot{x} \dot{+} \dot{y}')' \overset{\mathrm{p}}{=} (\dot{x} \dot{+} \dot{y})''$;

L14: MP$'$ $\rhd$ L12 $\rhd$ L13 $\gg$ $\dot{x}' \dot{+} \dot{y} \overset{\mathrm{p}}{=} (\dot{x} \dot{+} \dot{y})' \dot{\Rightarrow} \dot{x}' \dot{+} \dot{y}' \overset{\mathrm{p}}{=} (\dot{x} \dot{+} \dot{y}')'$;

L15: Gen$'$ $\rhd$ L14 $\gg$ $\dot{\forall} \dot{y} \colon (\dot{x}' \dot{+} \dot{y} \overset{\mathrm{p}}{=} (\dot{x} \dot{+} \dot{y})' \dot{\Rightarrow} \dot{x}' \dot{+} \dot{y}' \overset{\mathrm{p}}{=} (\dot{x} \dot{+} \dot{y}')')$ □

6.8.3 M Proposition 3.2(g)

We now have the necessary means to prove [M Proposition 3.2(g)].

[M Proposition 3.2(g)][19]

[S$'$ **lemma** M Proposition 3.2(g): $\dot{\forall} \dot{y} \colon \dot{x}' \dot{+} \dot{y} \overset{\mathrm{p}}{=} (\dot{x} \dot{+} \dot{y})'$]

S$'$ **proof of** M Proposition 3.2(g):

[19] [M Proposition 3.2(g) $\overset{\mathrm{pyk}}{=}$ "mendelson proposition three two g"]

L01:	S9$'$ $\gg$	$\dot{\forall}\dot{x}\colon(\dot{x}'\dot{+}\dot{0} \overset{p}{=} (\dot{x}\dot{+}\dot{0})') \dot{\Rightarrow}$	
		$\dot{\forall}\dot{y}\colon(\dot{x}'\dot{+}\dot{y} \overset{p}{=} (\dot{x}\dot{+}\dot{y})' \dot{\Rightarrow}$	
		$\dot{x}'\dot{+}\dot{y}' \overset{p}{=} (\dot{x}\dot{+}\dot{y}')') \dot{\Rightarrow}$	
		$\dot{\forall}\dot{y}\colon(\dot{x}'\dot{+}\dot{y} \overset{p}{=} (\dot{x}\dot{+}\dot{y})')$	;
L02:	M Proposition 3.2(g) (i) $\gg$	$\dot{\forall}\dot{x}\colon(\dot{x}'\dot{+}\dot{0} \overset{p}{=} (\dot{x}\dot{+}\dot{0})')$	;
L03:	MP$'$ $\rhd$ L01 $\rhd$ L02 $\gg$	$\dot{\forall}\dot{y}\colon(\dot{x}'\dot{+}\dot{y} \overset{p}{=} (\dot{x}\dot{+}\dot{y})' \dot{\Rightarrow}$	
		$\dot{x}'\dot{+}\dot{y}' \overset{p}{=} (\dot{x}\dot{+}\dot{y}')') \dot{\Rightarrow}$	
		$\dot{\forall}\dot{y}\colon(\dot{x}'\dot{+}\dot{y} \overset{p}{=} (\dot{x}\dot{+}\dot{y})')$	;
L04:	M Proposition 3.2(g) (ii) $\gg$	$\dot{\forall}\dot{y}\colon(\dot{x}'\dot{+}\dot{y} \overset{p}{=} (\dot{x}\dot{+}\dot{y})' \dot{\Rightarrow}$	
		$\dot{x}'\dot{+}\dot{y}' \overset{p}{=} (\dot{x}\dot{+}\dot{y}')')$	;
L05:	MP$'$ $\rhd$ L03 $\rhd$ L04 $\gg$	$\dot{\forall}\dot{y}\colon(\dot{x}'\dot{+}\dot{y} \overset{p}{=} (\dot{x}\dot{+}\dot{y})')$	$\square$

We have thus proved [M Proposition 3.2(g)].

6.9 M Proposition 3.2(h)

We now have the necessary means to show our objective, namely [M Proposition 3.2(h)]. As was the case for [M Proposition 3.2(f)] and [M Proposition 3.2(g)], this proposition will also be shown by induction, that is, with a base case and an inductive step but without an induction theorem, cf. refsection:3.2f.

6.9.1 M Proposition 3.2(h)(i)

The proof of [M Proposition 3.2(h)] makes use of the @ operator, which is defined on the base page, Section 6.3.9. It works as a quantifier eliminator, that is, a sort of reverse of generalization. First, we prove the base case.

[M Proposition 3.2(h)(i)][20]

[S$'$ **lemma** M Proposition 3.2(h)(i): $\dot{\forall}\dot{x}\colon(\dot{x}\dot{+}\dot{0} \overset{p}{=} \dot{0}\dot{+}\dot{x})$]

S$'$ **proof of** M Proposition 3.2(h)(i):

L01:	S5$'$ $\gg$	$\dot{x}\dot{+}\dot{0} \overset{p}{=} \dot{x}$	;
L02:	M Proposition 3.2(f) $\gg$	$\dot{\forall}\dot{x}\colon(\dot{x} \overset{p}{=} \dot{0}\dot{+}\dot{x})$	;
L03:	A4$'$ @ $\dot{x}$ $\gg$	$\dot{\forall}\dot{x}\colon(\dot{x} \overset{p}{=} \dot{0}\dot{+}\dot{x}) \dot{\Rightarrow} \dot{x} \overset{p}{=} \dot{0}\dot{+}\dot{x}$	;
L04:	MP$'$ $\rhd$ L03 $\rhd$ L02 $\gg$	$\dot{x} \overset{p}{=} \dot{0}\dot{+}\dot{x}$	;
L05:	M Proposition 3.2(c) $\gg$	$\dot{x}\dot{+}\dot{0} \overset{p}{=} \dot{x} \dot{\Rightarrow} (\dot{x} \overset{p}{=} \dot{0}\dot{+}\dot{x} \dot{\Rightarrow}$	
		$\dot{x}\dot{+}\dot{0} \overset{p}{=} \dot{0}\dot{+}\dot{x})$	;
L06:	MP$'$ $\rhd$ L05 $\rhd$ L01 $\gg$	$\dot{x} \overset{p}{=} \dot{0}\dot{+}\dot{x} \dot{\Rightarrow} \dot{x}\dot{+}\dot{0} \overset{p}{=} \dot{0}\dot{+}\dot{x}$	;
L07:	MP$'$ $\rhd$ L06 $\rhd$ L04 $\gg$	$\dot{x}\dot{+}\dot{0} \overset{p}{=} \dot{0}\dot{+}\dot{x}$	;
L08:	Gen$'$ $\rhd$ L07 $\gg$	$\dot{\forall}\dot{x}\colon(\dot{x}\dot{+}\dot{0} \overset{p}{=} \dot{0}\dot{+}\dot{x})$	$\square$

[20] [M Proposition 3.2(h)(i) $\overset{\text{pyk}}{=}$ "mendelson proposition three two h i"]

Thus, we have proved the base case and now proceed to the induction step. But before we can prove this we need a lemma, which must be proved first.

6.9.2 M Proposition 3.2(h)$_{(g)(ii)}$

In the proof of [M Proposition 3.2(h)(ii)] line 2 we see that [M Proposition 3.2(g)] is used. Cf. [4], p. 158. In comparison to the proof given here the x and y variables are swapped. We thus need a lemma allowing this version of the proposition. This is because we have used constants in stating and proving the proposition. The lemma we need is as follows:

[M Proposition 3.2(h)$_{(g)(ii)}$][21]

[S′ **lemma** M Proposition 3.2(h)$_{(g)(ii)}$: $\dot{\forall}\dot{y}\colon (\dot{x}' \dotplus \dot{y} \overset{p}{=} (\dot{x} \dotplus \dot{y})') \vdash \dot{y}' \dotplus \dot{x} \overset{p}{=} (\dot{y} \dotplus \dot{x})'$]

S′ **proof of** M Proposition 3.2(h)$_{(g)(ii)}$:

L01:	Premise $\gg$	$\dot{\forall}\dot{y}\colon (\dot{x}' \dotplus \dot{y} \overset{p}{=} (\dot{x} \dotplus \dot{y})')$	;
L02:	A4′ @ $\dot{z}$ $\gg$	$\dot{\forall}\dot{y}\colon (\dot{x}' \dotplus \dot{y} \overset{p}{=} (\dot{x} \dotplus \dot{y})') \dot{\Rightarrow} \dot{x}' \dotplus \dot{z} \overset{p}{=} (\dot{x} \dotplus \dot{z})'$	;
L03:	MP′ $\rhd$ L02 $\rhd$ L01 $\gg$	$\dot{x}' \dotplus \dot{z} \overset{p}{=} (\dot{x} \dotplus \dot{z})'$	;
L04:	Gen′ $\rhd$ L03 $\gg$	$\dot{\forall}\dot{x}\colon (\dot{x}' \dotplus \dot{z} \overset{p}{=} (\dot{x} \dotplus \dot{z})')$	;
L05:	A4′ @ $\dot{y}$ $\gg$	$\dot{\forall}\dot{x}\colon (\dot{x}' \dotplus \dot{z} \overset{p}{=} (\dot{x} \dotplus \dot{z})') \dot{\Rightarrow} (\dot{y}' \dotplus \dot{z} \overset{p}{=} (\dot{y} \dotplus \dot{z})')$	;
L06:	MP′ $\rhd$ L05 $\rhd$ L04 $\gg$	$\dot{y}' \dotplus \dot{z} \overset{p}{=} (\dot{y} \dotplus \dot{z})'$	;
L07:	Gen′ $\rhd$ L06 $\gg$	$\dot{\forall}\dot{z}\colon (\dot{y}' \dotplus \dot{z} \overset{p}{=} (\dot{y} \dotplus \dot{z})')$	;
L08:	A4′ @ $\dot{x}$ $\gg$	$\dot{\forall}\dot{z}\colon (\dot{y}' \dotplus \dot{z} \overset{p}{=} (\dot{y} \dotplus \dot{z})') \dot{\Rightarrow} \dot{y}' \dotplus \dot{x} \overset{p}{=} (\dot{y} \dotplus \dot{x})'$	;
L09:	MP′ $\rhd$ L08 $\rhd$ L07 $\gg$	$\dot{y}' \dotplus \dot{x} \overset{p}{=} (\dot{y} \dotplus \dot{x})'$	$\square$

We now have the necessary means to prove the induction step of [M Proposition 3.2(h)(ii)].

6.9.3 M Proposition 3.2(h)(ii)

We then prove the induction step.

[M Proposition 3.2(h)(ii)][22]

[S′ **lemma** M Proposition 3.2(h)(ii): $\dot{\forall}\dot{y}\colon (\dot{x} \dotplus \dot{y} \overset{p}{=} \dot{y} \dotplus \dot{x} \dot{\Rightarrow} \dot{x} \dotplus \dot{y}' \overset{p}{=} \dot{y}' \dotplus \dot{x})$]

[21] [M Proposition 3.2(h)$_{(g)(ii)}$ $\overset{\text{pyk}}{=}$ "mendelson proposition three two h g ii"]

[22] [M Proposition 3.2(h)(ii) $\overset{\text{pyk}}{=}$ "mendelson proposition three two h ii"]

S′ **proof of** M Proposition 3.2(h)(ii):

L01:	S6′ ≫	$\dot{x}\dot{+}\dot{y}' \overset{p}{=} (\dot{x}\dot{+}\dot{y})'$	;
L02:	M Proposition 3.2(g) ≫	$\dot{\forall}\dot{y}\colon (\dot{x}'\dot{+}\dot{y} \overset{p}{=} (\dot{x}\dot{+}\dot{y})')$	;
L03:	M Proposition 3.2(h)$_{(g)(ii)}$ ⊳ L02 ≫	$\dot{y}'\dot{+}\dot{x} \overset{p}{=} (\dot{y}\dot{+}\dot{x})'$	;
L04:	S2′ ≫	$\dot{x}\dot{+}\dot{y} \overset{p}{=} \dot{y}\dot{+}\dot{x} \dot{\Rightarrow} (\dot{x}\dot{+}\dot{y})' \overset{p}{=} (\dot{y}\dot{+}\dot{x})'$	;
L05:	M Proposition 3.2(c) ≫	$\dot{x}\dot{+}\dot{y}' \overset{p}{=} (\dot{x}\dot{+}\dot{y})' \dot{\Rightarrow} (\dot{x}\dot{+}\dot{y})' \overset{p}{=} (\dot{y}\dot{+}\dot{x})' \dot{\Rightarrow} \dot{x}\dot{+}\dot{y}' \overset{p}{=} (\dot{y}\dot{+}\dot{x})'$	;
L06:	M Tautology A ⊳ L05 ≫	$(\dot{x}\dot{+}\dot{y})' \overset{p}{=} (\dot{y}\dot{+}\dot{x})' \dot{\Rightarrow} \dot{x}\dot{+}\dot{y}' \overset{p}{=} (\dot{x}\dot{+}\dot{y})' \dot{\Rightarrow} \dot{x}\dot{+}\dot{y}' \overset{p}{=} (\dot{y}\dot{+}\dot{x})'$	;
L07:	M Tautology B ⊳ L04 ⊳ L06 ≫	$\dot{x}\dot{+}\dot{y} \overset{p}{=} \dot{y}\dot{+}\dot{x} \dot{\Rightarrow} \dot{x}\dot{+}\dot{y}' \overset{p}{=} (\dot{x}\dot{+}\dot{y})' \dot{\Rightarrow} \dot{x}\dot{+}\dot{y}' \overset{p}{=} (\dot{y}\dot{+}\dot{x})'$	;
L08:	M Tautology A ⊳ L07 ≫	$\dot{x}\dot{+}\dot{y}' \overset{p}{=} (\dot{x}\dot{+}\dot{y})' \dot{\Rightarrow} \dot{x}\dot{+}\dot{y} \overset{p}{=} \dot{y}\dot{+}\dot{x} \dot{\Rightarrow} \dot{x}\dot{+}\dot{y}' \overset{p}{=} (\dot{y}\dot{+}\dot{x})'$	;
L09:	MP′ ⊳ L08 ⊳ L01 ≫	$\dot{x}\dot{+}\dot{y} \overset{p}{=} \dot{y}\dot{+}\dot{x} \dot{\Rightarrow} \dot{x}\dot{+}\dot{y}' \overset{p}{=} (\dot{y}\dot{+}\dot{x})'$	;
L10:	M Proposition 3.2(d) ≫	$\dot{x}\dot{+}\dot{y}' \overset{p}{=} (\dot{y}\dot{+}\dot{x})' \dot{\Rightarrow} \dot{y}'\dot{+}\dot{x} \overset{p}{=} (\dot{y}\dot{+}\dot{x})' \dot{\Rightarrow} \dot{x}\dot{+}\dot{y}' \overset{p}{=} \dot{y}'\dot{+}\dot{x}$	;
L11:	M Tautology B ⊳ L09 ⊳ L10 ≫	$\dot{x}\dot{+}\dot{y} \overset{p}{=} \dot{y}\dot{+}\dot{x} \dot{\Rightarrow} \dot{y}'\dot{+}\dot{x} \overset{p}{=} (\dot{y}\dot{+}\dot{x})' \dot{\Rightarrow} \dot{x}\dot{+}\dot{y}' \overset{p}{=} \dot{y}'\dot{+}\dot{x}$	;
L12:	M Tautology A ⊳ L11 ≫	$\dot{y}'\dot{+}\dot{x} \overset{p}{=} (\dot{y}\dot{+}\dot{x})' \dot{\Rightarrow} \dot{x}\dot{+}\dot{y} \overset{p}{=} \dot{y}\dot{+}\dot{x} \dot{\Rightarrow} \dot{x}\dot{+}\dot{y}' \overset{p}{=} \dot{y}'\dot{+}\dot{x}$	;
L13:	MP′ ⊳ L12 ⊳ L03 ≫	$\dot{x}\dot{+}\dot{y} \overset{p}{=} \dot{y}\dot{+}\dot{x} \dot{\Rightarrow} \dot{x}\dot{+}\dot{y}' \overset{p}{=} \dot{y}'\dot{+}\dot{x}$	;
L14:	Gen′ ⊳ L13 ≫	$\dot{\forall}\dot{y}\colon (\dot{x}\dot{+}\dot{y} \overset{p}{=} \dot{y}\dot{+}\dot{x} \dot{\Rightarrow} \dot{x}\dot{+}\dot{y}' \overset{p}{=} \dot{y}'\dot{+}\dot{x})$	□

We have thus proved the inductive step and can proceed to our final objective.

6.9.4 M Proposition 3.2(h)

And finally, we can now prove the objective of this report, namely, the commutative law of addition.

[M Proposition 3.2(h)][23]

[S′ **lemma** M Proposition 3.2(h): $\dot{\forall}\dot{x}\colon \dot{\forall}\dot{y}\colon (\dot{x}\dot{+}\dot{y} \overset{p}{=} \dot{y}\dot{+}\dot{x})$]

S′ **proof of** M Proposition 3.2(h):

[23] [M Proposition 3.2(h) $\overset{\text{pyk}}{=}$ "mendelson proposition three two h"]

L01:	S9′ $\gg$	$\dot\forall \dot x\colon (\dot x \dot+ \dot 0 \overset{\mathrm{p}}{=} \dot 0 \dot+ \dot x) \dot\Rightarrow ((\dot\forall \dot y\colon (\dot x \dot+ \dot y \overset{\mathrm{p}}{=} \dot y \dot+ \dot x \dot\Rightarrow \dot x \dot+ \dot y' \overset{\mathrm{p}}{=} \dot y' \dot+ \dot x)) \dot\Rightarrow \dot\forall \dot y\colon (\dot x \dot+ \dot y \overset{\mathrm{p}}{=} \dot y \dot+ \dot x))$	;
L02:	M Proposition 3.2(h)(i) $\gg$	$\dot\forall \dot x\colon (\dot x \dot+ \dot 0 \overset{\mathrm{p}}{=} \dot 0 \dot+ \dot x)$	;
L03:	MP′ $\rhd$ L01 $\rhd$ L02 $\gg$	$\dot\forall \dot y\colon (\dot x \dot+ \dot y \overset{\mathrm{p}}{=} \dot y \dot+ \dot x \dot\Rightarrow \dot x \dot+ \dot y' \overset{\mathrm{p}}{=} \dot y' \dot+ \dot x) \dot\Rightarrow \dot\forall \dot y\colon (\dot x \dot+ \dot y \overset{\mathrm{p}}{=} \dot y \dot+ \dot x)$	;
L04:	M Proposition 3.2(h)(ii) $\gg$	$\dot\forall \dot y\colon (\dot x \dot+ \dot y \overset{\mathrm{p}}{=} \dot y \dot+ \dot x \dot\Rightarrow \dot x \dot+ \dot y' \overset{\mathrm{p}}{=} \dot y' \dot+ \dot x)$	;
L05:	MP′ $\rhd$ L03 $\rhd$ L04 $\gg$	$\dot\forall \dot y\colon (\dot x \dot+ \dot y \overset{\mathrm{p}}{=} \dot y \dot+ \dot x)$	;
L06:	Gen′ $\rhd$ L05 $\gg$	$\dot\forall \dot x\colon \dot\forall \dot y\colon (\dot x \dot+ \dot y \overset{\mathrm{p}}{=} \dot y \dot+ \dot x)$	$\square$

We have thus proved M Proposition 3.2(h), that is, 1 stating the commutative law of addition. We have done this using variables x and y, but this does not make a difference. Cf. 6.7.3.

7 Conclusion

In this report we have stated and proved a number of propositions from [4], p. 156. More precisely we have proved Proposition 3.2(a)-(h), except Proposition 3.2(e). The proofs have been carried out with the ambition of keeping things as simple as possible. We have succeded in doing so, proving all propositions by the help of the theorems given in [**Theory** S′] defined on the peano page, two tautologies, and one lemma needed in the proof of our final theorem. The final theorem stated our original assignment, namely the commutativity of addition. We can thus conclude that we have solved the assignment properly.

Furthermore, for the inexperienced user and logician we have intended to give a short introduction to the Logiweb system, axiomatic theories, and peano arithmetic. The introduction should make the report self-contained, though we have given references to relevant literature where needed.

A The Name of the Page

This defines the name of the page:

[peano commutativity $\overset{\text{pyk}}{=}$ "peano commutativity"]

B Priority Table

Priority table
Preassociative
[peano commutativity], [base], [bracket $*$ end bracket], [big bracket $*$ end bracket], [math $*$ end math], [**flush left** [$*$]], [x], [y], [z], [[$* \bowtie *$]], [[$* \overset{*}{\rightarrow} *$]], [pyk], [tex], [name], [prio], [$*$], [T], [if($*, *, *$)], [[$* \overset{*}{\Rightarrow} *$]], [val], [claim], [$\bot$], [f($*$)], [$(*)^{\mathrm{I}}$], [$\mathsf{F}$], [$\underline{0}$], [$\underline{1}$], [$\underline{2}$], [$\underline{3}$], [$\underline{4}$], [$\underline{5}$], [$\underline{6}$], [$\underline{7}$], [$\underline{8}$], [$\underline{9}$], [0], [1], [2], [3], [4], [5], [6], [7], [8], [9], [a], [b], [c], [d], [e], [f], [g], [h], [i], [j], [k], [l], [m], [n], [o], [p], [q], [r], [s], [t], [u], [v], [w], [$(*)^{\mathrm{M}}$], [If($*, *, *$)], [array{$*$} $*$ end array], [l], [c], [r], [empty], [$\langle * \mid * := * \rangle$], [$\mathcal{M}(*)$], [$\tilde{\mathcal{U}}(*)$], [$\mathcal{U}(*)$], [$\mathcal{U}^{\mathrm{M}}(*)$], [**apply**($*, *$)], [$\mathbf{apply}_1(*, *)$], [identifier($*$)], [$\mathrm{identifier}_1(*, *)$], [array-plus($*, *$)], [array-remove($*, *, *$)], [array-put($*, *, *, *$)], [array-add($*, *, *, *, *$)], [bit($*, *$)], [$\mathrm{bit}_1(*, *)$], [rack], ["vector"], ["bibliography"], ["dictionary"], ["body"], ["codex"], ["expansion"], ["code"], ["cache"], ["diagnose"], ["pyk"], ["tex"], ["texname"], ["value"], ["message"], ["macro"], ["definition"], ["unpack"], ["claim"], ["priority"], ["lambda"], ["apply"], ["true"], ["if"], ["quote"], ["proclaim"], ["define"], ["introduce"], ["hide"], ["pre"], ["post"], [$\mathcal{E}(*, *, *)$], [$\mathcal{E}_2(*, *, *, *, *)$], [$\mathcal{E}_3(*, *, *, *)$], [$\mathcal{E}_4(*, *, *, *)$], [**lookup**($*, *, *$)], [**abstract**($*, *, *, *$)], [[$\lceil * \rceil$]], [$\mathcal{M}(*, *, *)$], [$\mathcal{M}_2(*, *, *, *)$], [$\mathcal{M}^*(*, *, *)$], [macro], [$\mathrm{s}_0$], [**zip**($*, *$)], [$\mathbf{assoc}_1(*, *, *)$], [$(*)^{\mathbf{p}}$], [self], [[$* \ddot{=} *$]], [[$* \doteq *$]], [[$* \acute{=} *$]], [[$* \overset{\text{pyk}}{=} *$]], [[$* \overset{\text{tex}}{=} *$]], [[$* \overset{\text{name}}{=} *$]], [**Priority table**[$*$]], [$\tilde{\mathcal{M}}_1$], [$\tilde{\mathcal{M}}_2(*)$], [$\tilde{\mathcal{M}}_3(*)$], [$\tilde{\mathcal{M}}_4(*, *, *, *)$], [$\tilde{\mathcal{M}}(*, *, *)$], [$\tilde{\mathcal{Q}}(*, *, *)$], [$\tilde{\mathcal{Q}}_2(*, *, *)$], [$\tilde{\mathcal{Q}}_3(*, *, *, *)$], [$\tilde{\mathcal{Q}}^*(*, *, *)$], [$(*)$], [**aspect**($*, *$)], [**aspect**($*, *, *$)], [$\langle * \rangle$], [$\mathbf{tuple}_1(*)$], [$\mathbf{tuple}_2(*)$], [$\mathrm{let}_2(*, *)$], [$\mathrm{let}_1(*, *)$], [[$* \overset{\text{claim}}{=} *$]], [checker], [**check**($*, *$)], [$\mathbf{check}_2(*, *, *)$], [$\mathbf{check}_3(*, *, *)$], [$\mathbf{check}^*(*, *)$], [$\mathbf{check}^*_2(*, *, *)$], [$[*]^{\cdot}$], [$[*]^{-}$], [$[*]^{\circ}$], [msg], [[$* \overset{\text{msg}}{=} *$]], [<stmt>], [stmt], [[$* \overset{\text{stmt}}{=} *$]], [HeadNil′], [HeadPair′], [Transitivity′], [$\perp\!\!\!\perp$], [Contra′], [T'_{E}], [L_1], [$\underline{*}$], [$\mathcal{A}$], [$\mathcal{B}$], [$\mathcal{C}$], [$\mathcal{D}$], [$\mathcal{E}$], [$\mathcal{F}$], [$\mathcal{G}$], [$\mathcal{H}$], [$\mathcal{I}$], [$\mathcal{J}$], [$\mathcal{K}$], [$\mathcal{L}$], [$\mathcal{M}$], [$\mathcal{N}$], [$\mathcal{O}$], [$\mathcal{P}$], [$\mathcal{Q}$], [$\mathcal{R}$], [$\mathcal{S}$], [$\mathcal{T}$], [$\mathcal{U}$], [$\mathcal{V}$], [$\mathcal{W}$], [$\mathcal{X}$], [$\mathcal{Y}$], [$\mathcal{Z}$], [$\langle * \mid * := * \rangle$], [$\langle^* * \mid * := * \rangle$], [$\emptyset$], [Remainder], [$(*)^{\mathbf{v}}$], [intro($*, *, *, *$)], [intro($*, *, *$)], [error($*, *$)], [$\mathrm{error}_2(*, *)$], [proof($*, *, *$)], [$\mathrm{proof}_2(*, *)$], [$\mathcal{S}(*, *)$], [$\mathcal{S}^{\mathrm{I}}(*, *)$], [$\mathcal{S}^{\rhd}(*, *)$], [$\mathcal{S}^{\rhd}_1(*, *, *)$], [$\mathcal{S}^{\mathrm{E}}(*, *)$], [$\mathcal{S}^{\mathrm{E}}_1(*, *, *)$], [$\mathcal{S}^{+}(*, *)$], [$\mathcal{S}^{+}_1(*, *, *)$], [$\mathcal{S}^{-}(*, *)$], [$\mathcal{S}^{-}_1(*, *, *)$], [$\mathcal{S}^{*}(*, *)$], [$\mathcal{S}^{*}_1(*, *, *)$], [$\mathcal{S}^{*}_2(*, *, *, *)$], [$\mathcal{S}^{@}(*, *)$], [$\mathcal{S}^{@}_1(*, *, *)$], [$\mathcal{S}^{\vdash}(*, *)$], [$\mathcal{S}^{\vdash}_1(*, *, *, *)$], [$\mathcal{S}^{\Vdash}(*, *)$], [$\mathcal{S}^{\Vdash}_1(*, *, *, *)$], [$\mathcal{S}^{\mathrm{i.e.}}(*, *)$], [$\mathcal{S}^{\mathrm{i.e.}}_1(*, *, *, *)$], [$\mathcal{S}^{\mathrm{i.e.}}_2(*, *, *, *, *)$], [$\mathcal{S}^{\forall}(*, *)$], [$\mathcal{S}^{\forall}_1(*, *, *, *)$], [$\mathcal{S}^{;}(*, *)$], [$\mathcal{S}^{;}_1(*, *, *)$], [$\mathcal{S}^{;}_2(*, *, *, *)$], [$\mathcal{T}(*)$], [claims($*, *, *$)], [$\mathrm{claims}_2(*, *, *)$], [<proof>], [proof], [[**Lemma** $*$: $*$]], [[**Proof of** $*$: $*$]], [[$*$ **lemma** $*$: $*$]], [[$*$ **antilemma** $*$: $*$]], [[$*$ **rule** $*$: $*$]], [[$*$ **antirule** $*$: $*$]], [verifier], [$\mathcal{V}_1(*)$], [$\mathcal{V}_2(*, *)$], [$\mathcal{V}_3(*, *, *, *)$], [$\mathcal{V}_4(*, *)$], [$\mathcal{V}_5(*, *, *, *)$], [$\mathcal{V}_6(*, *, *, *)$], [$\mathcal{V}_7(*, *, *, *)$], [Cut($*, *$)], [$\mathrm{Head}_{\oplus}(*)$], [$\mathrm{Tail}_{\oplus}(*)$], [$\mathrm{rule}_1(*, *)$], [rule($*, *$)],

[Rule tactic], [Plus(∗, ∗)], [[**Theory** ∗]], [theory$_2$(∗, ∗)], [theory$_3$(∗, ∗)], [theory$_4$(∗, ∗, ∗)], [HeadNil″], [HeadPair″], [Transitivity″], [Contra″], [HeadNil], [HeadPair], [Transitivity], [Contra], [T$_\mathrm{E}$], [ragged right], [ragged right expansion], [parm(∗, ∗, ∗)], [parm*(∗, ∗, ∗)], [inst(∗, ∗)], [inst*(∗, ∗)], [occur(∗, ∗, ∗)], [occur*(∗, ∗, ∗)], [unify(∗ = ∗, ∗)], [unify*(∗ = ∗, ∗)], [unify$_2$(∗ = ∗, ∗)], [L$_\mathrm{a}$], [L$_\mathrm{b}$], [L$_\mathrm{c}$], [L$_\mathrm{d}$], [L$_\mathrm{e}$], [L$_\mathrm{f}$], [L$_\mathrm{g}$], [L$_\mathrm{h}$], [L$_\mathrm{i}$], [L$_\mathrm{j}$], [L$_\mathrm{k}$], [L$_\mathrm{l}$], [L$_\mathrm{m}$], [L$_\mathrm{n}$], [L$_\mathrm{o}$], [L$_\mathrm{p}$], [L$_\mathrm{q}$], [L$_\mathrm{r}$], [L$_\mathrm{s}$], [L$_\mathrm{t}$], [L$_\mathrm{u}$], [L$_\mathrm{v}$], [L$_\mathrm{w}$], [L$_\mathrm{x}$], [L$_\mathrm{y}$], [L$_\mathrm{z}$], [L$_\mathrm{A}$], [L$_\mathrm{B}$], [L$_\mathrm{C}$], [L$_\mathrm{D}$], [L$_\mathrm{E}$], [L$_\mathrm{F}$], [L$_\mathrm{G}$], [L$_\mathrm{H}$], [L$_\mathrm{I}$], [L$_\mathrm{J}$], [L$_\mathrm{K}$], [L$_\mathrm{L}$], [L$_\mathrm{M}$], [L$_\mathrm{N}$], [L$_\mathrm{O}$], [L$_\mathrm{P}$], [L$_\mathrm{Q}$], [L$_\mathrm{R}$], [L$_\mathrm{S}$], [L$_\mathrm{T}$], [L$_\mathrm{U}$], [L$_\mathrm{V}$], [L$_\mathrm{W}$], [L$_\mathrm{X}$], [L$_\mathrm{Y}$], [L$_\mathrm{Z}$], [L$_?$], [Reflexivity], [Reflexivity$_1$], [Commutativity], [Commutativity$_1$], [<tactic>], [tactic], [[∗ $\overset{\text{tactic}}{=}$ ∗]], [$\mathcal{P}$(∗, ∗, ∗)], [$\mathcal{P}^*$(∗, ∗, ∗)], [p$_0$], [conclude$_1$(∗, ∗)], [conclude$_2$(∗, ∗, ∗)], [conclude$_3$(∗, ∗, ∗, ∗)], [conclude$_4$(∗, ∗)], [peano], [$\dot{0}$], [$\dot{1}$], [$\dot{2}$], [$\dot{a}$], [$\dot{b}$], [$\dot{c}$], [$\dot{d}$], [$\dot{e}$], [$\dot{f}$], [$\dot{g}$], [$\dot{h}$], [$\dot{i}$], [$\dot{j}$], [$\dot{k}$], [$\dot{l}$], [$\dot{m}$], [$\dot{n}$], [$\dot{o}$], [$\dot{p}$], [$\dot{q}$], [$\dot{r}$], [$\dot{s}$], [$\dot{t}$], [$\dot{u}$], [$\dot{v}$], [$\dot{w}$], [$\dot{x}$], [$\dot{y}$], [$\dot{z}$], [nonfree(∗, ∗)], [nonfree*(∗, ∗)], [free⟨∗|∗ := ∗⟩], [free*⟨∗|∗ := ∗⟩], [∗≡⟨∗|∗ := ∗⟩], [∗≡⟨*∗|∗ := ∗⟩], [S], [A1], [A2], [A3], [A4], [A5], [S1], [S2], [S3], [S4], [S5], [S6], [S7], [S8], [S9], [MP], [Gen], [L3.2(a)], [S′], [A1′], [A2′], [A3′], [A4′], [A5′], [S1′], [S2′], [S3′], [S4′], [S5′], [S6′], [S7′], [S8′], [S9′], [MP′], [Gen′], [L3.2(a)′], [M1.7], [MP$'_\mathrm{h}$], [Hypothesize], [Gen$'_\mathrm{h}$], [M3.2(a)], [M3.2(a)$_\mathrm{h}$], [M3.2(b)$_\mathrm{h}$], [M3.1(S1′)$_\mathrm{h}$], [M3.2(c)$_\mathrm{h}$], [M3.1(S2′)$_\mathrm{h}$], [M3.1(S5′)$_\mathrm{h}$], [M3.1(S6′)$_\mathrm{h}$], [M3.2(f)], [M Lemma 1.8], [M Proposition 3.2(a)], [M Proposition 3.2(b)], [M Proposition 3.2(c)], [M Proposition 3.2(d)], [M Proposition 3.2(f) (i)], [M Proposition 3.2(f) (ii)], [M Proposition 3.2(f)], [M Proposition 3.2(g) (i)], [M Proposition 3.2(g) (ii)], [M Proposition 3.2(g)], [M Proposition 3.2(h)(i)], [M Proposition 3.2(h)$_\mathrm{(g)(ii)}$], [M Proposition 3.2(h)(ii)], [M Proposition 3.2(h)], [M Tautology A], [M Tautology B];

Preassociative

[∗_{∗}], [∗′], [∗[∗]], [∗[∗→∗]], [∗[∗⇒∗]], [$\dot{∗}$];

Preassociative

[“ ∗ ”], [], [(∗)$^\mathbf{t}$], [string(∗) + ∗], [string(∗) ++ ∗], [
∗], [∗], [!∗], ["∗], [#∗], [$∗], [%∗], [&∗], ['∗], [(∗], [)∗], [∗∗], [+∗], [,∗], [-∗], [.∗], [/∗], [0∗], [1∗], [2∗], [3∗], [4∗], [5∗], [6∗], [7∗], [8∗], [9∗], [:∗], [;∗], [<∗], [=∗], [>∗], [?∗], [@∗], [A∗], [B∗], [C∗], [D∗], [E∗], [F∗], [G∗], [H∗], [I∗], [J∗], [K∗], [L∗], [M∗], [N∗], [O∗], [P∗], [Q∗], [R∗], [S∗], [T∗], [U∗], [V∗], [W∗], [X∗], [Y∗], [Z∗], [[∗], [\∗], []∗], [^∗], [_∗], [‘∗], [a∗], [b∗], [c∗], [d∗], [e∗], [f∗], [g∗], [h∗], [i∗], [j∗], [k∗], [l∗], [m∗], [n∗], [o∗], [p∗], [q∗], [r∗], [s∗], [t∗], [u∗], [v∗], [w∗], [x∗], [y∗], [z∗], [{∗], [|∗], [}∗], [~∗], [**Preassociative** ∗; ∗], [**Postassociative** ∗; ∗], [[∗], ∗], [priority ∗ end], [newline ∗], [macro newline ∗];

Preassociative

[∗0], [∗1], [0b], [∗-color(∗)], [∗-color*(∗)];

Preassociative

[∗ ’ ∗], [∗ ‘ ∗];

Preassociative

[$∗^\mathrm{H}$], [$∗^\mathrm{T}$], [$∗^\mathrm{U}$], [$∗^\mathrm{h}$], [$∗^\mathrm{t}$], [$∗^\mathrm{s}$], [$∗^\mathrm{c}$], [$∗^\mathrm{d}$], [$∗^\mathrm{a}$], [$∗^\mathrm{C}$], [$∗^\mathrm{M}$], [$∗^\mathrm{B}$], [$∗^\mathrm{r}$], [$∗^\mathrm{i}$], [$∗^\mathrm{d}$], [$∗^\mathrm{R}$], [$∗^0$], [$∗^1$], [$∗^2$], [$∗^3$], [$∗^4$], [$∗^5$], [$∗^6$], [$∗^7$], [$∗^8$], [$∗^9$], [$∗^\mathrm{E}$], [$∗^\mathcal{V}$], [$∗^\mathcal{C}$], [$∗^{\mathcal{C}^*}$], [∗′];

Preassociative

$[* \cdot *], [* \cdot_0 *], [* \mathbin{\dot{\cdot}} *];$

Preassociative

$[* + *], [* +_0 *], [* +_1 *], [* - *], [* -_0 *], [* -_1 *], [* \mathbin{\dot{+}} *];$

Preassociative

$[* \cup \{*\}], [* \cup *], [* \backslash \{*\}];$

Postassociative

$[* \therefore *], [* \underline{\therefore} *], [* \underline{::} *], [* \underline{+2*} *], [* :: *], [* +2* *];$

Postassociative

$[*, *];$

Preassociative

$[* \overset{\mathrm{B}}{\approx} *], [* \overset{\mathrm{D}}{\approx} *], [* \overset{\mathrm{C}}{\approx} *], [* \overset{\mathrm{P}}{\approx} *], [* \approx *], [* = *], [* \overset{+}{\rightarrow} *], [* \overset{\mathrm{t}}{=} *], [* \overset{\mathrm{t}^*}{=} *], [* \overset{\mathrm{r}}{=} *],$
$[* \in_{\mathrm{t}} *], [* \subseteq_{\mathrm{T}} *], [* \overset{\mathrm{T}}{=} *], [* \overset{\mathrm{s}}{=} *], [* \text{ free in } *], [* \text{ free in}^* *], [* \text{ free for } * \text{ in } *],$
$[* \text{ free for}^* * \text{ in } *], [* \in_{\mathrm{c}} *], [* < *], [* <' *], [* \leq' *], [* \overset{\mathrm{p}}{=} *], [*^{\mathcal{P}}];$

Preassociative

$[\neg *], [\dot{\neg} *];$

Preassociative

$[* \wedge *], [* \ddot{\wedge} *], [* \tilde{\wedge} *], [* \wedge_{\mathrm{c}} *], [* \dot{\wedge} *];$

Preassociative

$[* \vee *], [* \parallel *], [* \ddot{\vee} *], [* \dot{\vee} *];$

Preassociative

$[\dot{\forall} * \colon *], [\dot{\exists} * \colon *];$

Postassociative

$[* \mathrel{\ddot{\Rightarrow}} *], [* \mathrel{\dot{\Rightarrow}} *], [* \mathrel{\ddot{\Leftrightarrow}} *];$

Postassociative

$[* : *], [*!*];$

Preassociative

$[* \left\{ \begin{array}{c} * \\ * \end{array} \right.];$

Preassociative

$[\lambda * . *], [\Lambda *], [\textbf{if } * \textbf{ then } * \textbf{ else } *], [\textbf{let } * = * \textbf{ in } *], [\textbf{let } * \mathrel{\ddot{=}} * \text{ in } *];$

Preassociative

$[*^{\mathrm{I}}], [*^{\triangleright}], [*^{\mathrm{V}}], [*^{+}], [*^{-}], [*^{*}];$

Preassociative

$[* @ *], [* \rhd *], [* \mathrel{\blacktriangleright} *], [* \gg *];$

Postassociative

$[* \vdash *], [* \Vdash *], [* \text{ i.e. } *];$

Preassociative

$[\forall * \colon *];$

Postassociative

$[* \oplus *];$

Postassociative

$[*; *];$

Preassociative

$[* \text{ proves } *];$

Preassociative

$[* \textbf{ proof of } * : *], [\text{Line} * : * \gg *; *], [\text{Last line} * \gg * \square],$
$[\text{Line} * : \text{Premise} \gg *; *], [\text{Line} * : \text{Side-condition} \gg *; *], [\text{Arbitrary} \gg *; *],$
$[\text{Local} \gg * = *; *];$
Postassociative
$[* \text{ then } *], [*[\,*\,]*];$
Preassociative
$[*\&*];$
Preassociative
$[*\backslash\backslash*];$ **End table**

C Bibliography

[1] Klaus Grue. *Introduction to Logiweb*. http://www.diku.dk/~grue/logiweb/20050502/home/grue/check/GRD-2005-05-26-UTC-06-49-32-964854/, 2005.

[2] Klaus Grue. *A Logiweb base page*. http://www.diku.dk/~grue/logiweb/20050502/home/grue/base/latest/, 2005.

[3] Klaus Grue. *Peano arithmetic*. http://www.diku.dk/~grue/logiweb/20050502/home/grue/peano-axioms/latest/, 2005.

[4] E. Mendelson. *Introduction to Mathematical Logic*. Wadsworth and Brooks, 3. edition, 1987.