# A Philosophical Origin Story

Randall Holmes

September 14, 2020

We begin with ordinary naive discourse, pieces of ordinary language and expressions and equations of mathematics.

We start by saying that equations of mathematics and sentences of ordinary language are for us names of the truth values, the True, which we write **true** and the False, which we write **false**.

We need logical constructions prior to our metaphysical investigatiion. We provide a primitive $\neg$: $\neg p$ denotes the True if $p$ is **false** and the False otherwise. We provide the primitive $\rightarrow$ of material implication. $p \rightarrow q$ is the False if $p$ is the True and $\neg q$ is the True, and otherwise is the False.

We write $\vdash p$ to mean that we assert that $p$ is true. This will not appear as a proper component of any other expression.

For any expressions $p, q$ in our language, if we have $\vdash p$ and we have $\vdash p \rightarrow q$, we can get $\vdash q$. This is the rule of modus ponens.

Our entire work here is ultimately about the care and feeding of unknown letters such as $p$, $q$. We note that for the moment we are using them as a device of generality: anything may be substituted for them. In general if we assert $\vdash p$ and $p$ is actually an expression with unknowns in it, we are asserting it for all values of the unknowns.

An expression $p \rightarrow q \rightarrow r$ will be read as $p \rightarrow (q \rightarrow r)$.

We suggest some axioms. These are not the same as Frege's axioms, but they achieve the same purpose.

1. $\vdash x = x$: equality is reflexive.
2. $\vdash (x = y) \rightarrow (F(x) \rightarrow F(y))$: equality has the substitution property.
3. $\vdash F(\mathbf{true}) \rightarrow F(\mathbf{false}) \rightarrow F(\neg x)$: whatever is true of the True and the False is true of all propositions (any proposition can be written as a

negation, and any negation is a proposition). This is the poor man's implementation of completeness of propositional logic, supporting truth table reasoning.

4. $\vdash (p \to q) = \neg\neg(p \to q)$: implications are propositions.

5. $\vdash (p \to q) = (\neg q \to \neg p)$: the contrapositive, the point of which here is that components of propositions can be replaced with their double negations (apply it twice to see that).

6. $\vdash \neg\neg\neg p = \neg p$: double negation; negations are propositions.

7. $\vdash (x = y) = \neg\neg(x = y)$: equations are propositions.

8. $\vdash \mathbf{false} = \neg\mathbf{true}$: the truth table for negation (1).

9. $\vdash \mathbf{true} = \neg\mathbf{false}$: the truth table for negation (2).

10. $\vdash (\mathbf{true} \to p) = \neg\neg p$: the truth table for implication (1).

11. $\vdash (\mathbf{false} \to p) = \mathbf{true}$: the truth table for implication (2).

12. $\vdash (p = \mathbf{true}) = \neg\neg p$: double negation clarified.

We define $p \vee q$ as $(\neg p) \to q$ and $p \wedge q$ as $\neg(p \to \neg q)$. Equality works for the biconditional, though we could define it in the usual way: strictly speaking, $p \leftrightarrow q$ can be defined as $(\neg p) = \neg q$.

Now we introduce quantification $(\forall \mathtt{x} : P(\mathtt{x}))$. We have a rule of inference: from $\vdash P(x)$ we can deduce $\vdash (\forall \mathtt{x} : P(\mathtt{x}))$, for any bound variable $\mathtt{x}$ such that there is no occurrence of $\mathtt{x}$ in $P(x)$. We also always assume that $x$ occurs in an expression $P(x)$.

This exemplifies restrictions on variable binding. Bound variables of each type are distinct from free variables of that type (we use a distinct type face) and the scope of a binder over a given bound variable cannot include another binder over the same bound variable. In this way, both forms of variable capture are averted: substitution for an object variable, free or bound, is simply substitution in every context. Preventing pathologies of substitution is important in a treatment of functions "naively" as substitution contexts. Substitution for function variables is discussed below.

We introduce $\vdash (\forall \mathtt{x} : P(\mathtt{x})) \rightarrow P(a)$ as an axiom. Where assertions are made, they are made in full generality: $\vdash (\forall \mathtt{y} : P(\mathtt{y})) \rightarrow P(2+2)$ is a case of the just given axiom.

The use of function notation above is naive; explication of the naive notion of function is an essential part of the story. We are following the convention that any unknown object is a lower case letter and any unknown function is an upper case letter. $P(x)$, the predicate in the quantification definition, is a function whose values are truth values from our standpoint (a concept).

We also have quantification over functions. Notice that we try always to use lower case variables for object and upper case variables for functions.

We have a rule of inference: from $\vdash \mathcal{P}(F)$ we can deduce $\vdash (\forall \mathtt{F} : \mathcal{P}(\mathtt{F}))$, where $\mathtt{F}$ is any bound function variable such that $\mathcal{P}(F)$ contains no occurrence of $\mathtt{F}$.

We introduce $\vdash (\forall \mathtt{F} : \mathcal{P}(\mathtt{F})) \rightarrow \mathcal{P}(G)$ as an axiom.

A letter $\mathcal{P}$ stands for a function taking first level functions such as $F$ (one argument functions taking an object argument to an object value) to objects. The practical effect is that any sentence or expression involving a function variable $F$ can be read $\mathcal{P}(F)$ in the same way that we are reading each general sentence or expression involving an object variable $x$ as $F(x)$. There will be no bound variables of this kind.

We state an axiom governing equality:

$$(x = y) = (\forall \mathtt{F} : \mathtt{F}(y) \rightarrow \mathtt{F}(x)).$$

This might actually be a theorem in my somewhat different presentation of Frege's version of logic, but I am not concerned to show this.

The existential quantifiers $(\exists \mathtt{x} : F(\mathtt{x}))$ and $(\exists \mathtt{F} : \mathcal{P}(\mathtt{F}))$ can be defined as $\neg(\forall \mathtt{x} : \neg F(\mathtt{x}))$ and $\neg(\forall \mathtt{F} : \neg \mathcal{P}(\mathtt{F}))$

Now we discuss functions. We are viewing a (first level) function as an entity determined by an object expression with a gap in it, as for example $f(x) = x^2 + x + 1$. But Frege warns us against confusing an expression $x^2 + x + 1$ for an unknown number with a function. Consider for example an expression $x^2 + a$. Is this a function sending $x$ to $x^2 + a$ for a parameter $a$ or a function sending $a$ to $x^2 + a$ for a parameter $x$, or is it a function of two variables? Frege proposes a special symbol for the input variable of a function: we use $\mathbf{X}$ for this purpose (and $\mathbf{Y}$ for a second argument). Using this notation, we can distinguish $\mathbf{X}^2 + a$, $x^2 + \mathbf{X}$ and $\mathbf{X}^2 + \mathbf{Y}$. Our original example is written $\mathbf{X}^2 + \mathbf{X} + 1$. An oddity of this sort of notation (which

Russell also had to deal with) is that it has no markings of scope. Frege never writes an expression of this kind as a component of another expression. The only proper subterms of terms of Frege's language which represent functions are variables (with a technical exception which we shortly note, which I do not think occurs in Frege). But we do know how to substitute an expression such as $\mathbf{X}^2 + a$ for a function letter variable $F$: the result of replacing $F$ with $\mathbf{X}^2 + a$ in $F(2)$ is $2^2 + a$. The only contexts in which a function variable can appear in our language are in applied position or as an argument of a variable $\mathcal{F}$ of second level. We adopt the convention that (for example) $\mathcal{F}(\mathbf{X}^2 + a)$ is a permitted notation; Frege does not have to deal with this situation that I am aware of, but Russell does have a few examples of propositional function notations appearing as arguments. Notice that when all second level variables are cashed out (as we should expect them to be: there are no quantifiers over these and they are introduced only for purposes of second-order quantification), then substitution will eliminate all complex function terms. The only place in which a complex function term can appear as a proper component of a term is as an argument of a second level function variable.

Complex second level function terms can be built using notations $\mathbf{F}$ and $\mathbf{G}$ for a first and second function argument. We do not think $\mathbf{G}$ is needed, but we might as well provide it. A second level variable (such as $\mathcal{P}$) appears *only* in applied position, so replacement of a second level function variable with a complex second level function is straightforward and always completely eliminates expressions in $\mathbf{F}$ and $\mathbf{G}$. For example, the universal quantifier is the second level function $(\forall \mathtt{x} : \mathbf{F}(\mathtt{x}))$. Replacing $\mathcal{P}$ with $(\forall \mathtt{x} : \mathbf{F}(\mathtt{x}))$ and $F$ with $\mathbf{X} = \mathbf{X}$ in $\mathcal{P}(F)$ gives $(\forall \mathtt{x} : \mathbf{F}\mathtt{x}))$ then $(\forall \mathtt{x} : \mathtt{x} = \mathtt{x})$ or else (stranger) $\mathcal{P}(\mathbf{X} = \mathbf{X}$ then $(\forall \mathtt{x} : \mathtt{x} = \mathtt{x})$; the final result is the same.

We make a side remark that expressions differing only by renaming of bound variables of course have the same meaning, and sometimes in order to effect a substitution it is necessary to rename bound variables: for example, our rules do not permit us to directly apply $(\exists \mathtt{x} : \mathtt{x} = \mathbf{X})$ to $(\mathtt{x} \mapsto \mathtt{x} = \mathtt{x})$. In such a case, we can nonetheless execute the intended function application by first renaming bound variables used in either the function or the object to which it is applied, as needed: the result here is something like $(\exists \mathtt{x} : \mathtt{x} = (\mathtt{u} \mapsto \mathtt{u} = \mathtt{u}))$ and we are indifferent to what new variable we adopt. This is a design choice: we could also allow substitution of any expression into any other, at the price of causing substitution of an expression for a bound variable $\mathtt{x}$ to be something other than direct substitution of text (the qualification being

that substitutions for $\mathtt{x}$ would have no effect on expressions in which $\mathtt{x}$ is bound).

Frege is a realist about functions. He quantifies over them, so he is committed to them as a domain of entities (appealing, anachronistically perhaps, to Quine). He denies that they are actually pieces of syntax (syntax gives us functions but not necessarily all of them). But he is careful to distinguish them from objects. I cannot say why he calls them "unsaturated", as this seems to be a property of them as syntax. But he is careful to draw the distinction of logical type between function and object (in which he is wise, though as it turned out not wise enough).

He does not propose identity criteria for functions. What he does do (and this is where, in spite of what he says, he departs from pure logic) is postulate objects, the "courses of values", correlated with the functions. Functions are created by a process of abstraction which may be regarded as purely logical. The postulation of courses of values is bringing abstractions down into the concrete.

We note that if $F$ represents $G(\mathbf{X})$ then $F(a)$ represents $G(a)$, and the change of letter from $\mathbf{X}$ to $a$ represents substitution of the term $a$ (which may be complex) for $\mathbf{X}$ in the term $G(\mathbf{X})$ (which may be complex and involve several occurrences of $\mathbf{X}$).

For each choice of bound variable $\mathtt{x}$ and expression $f(\mathbf{X})$ which includes no occurrence of $\mathtt{x}$, we provide notation $(\mathtt{x} \mapsto f(\mathtt{x}))$ for the course of values of the function $f(\mathbf{X})$. He postulates courses of values only for object-valued functions of a single object argument. Frege is even more careful about variable binding than we are, using different series of variables for courses of values and instances of quantification over objects. We see no reason to do this.

The axiom expressing his postulation of courses of values is this:

$$(\forall \mathtt{F} : (\forall \mathtt{G} : (\forall \mathtt{x} : \mathtt{F}(\mathtt{x}) = \mathtt{G}(\mathtt{x})) = ((\mathtt{u} \mapsto \mathtt{F}(\mathtt{u})) = (\mathtt{v} \mapsto \mathtt{G}(\mathtt{v})))))$$

This says that two functions have the same course of values if and only if they have the same extension. We mix up the bound variables to make it clear that bound variable renaming does not affect the identity of a course of values.

We define application of a course of values, considered as representing a function, to an object. We use Russell's notation $f`x$ for this.

$y = f`x$ is to hold if $(\exists \mathtt{F} : f = (\mathtt{x} \mapsto \mathtt{F}(\mathtt{x})) \wedge y = \mathtt{F}(x))$.

This illustrates the need for the last ingredient of Frege's system. We provide a function $\backslash$ which satisfies the axiom $\vdash a = \backslash(\mathtt{x} \mapsto \mathtt{x} = a)$. In other words, $\backslash f$, if $f$ is the course of values of a concept, is the unique $a$ such that $\mathtt{true} = f\text{`}a$, if there is one. This serves the same purpose as Russell's definite description operator. Frege says things about the value of $\backslash f$ in unintended cases, but his axioms do not address this.

And this allows us to write

$f\text{`}x = \backslash(\mathtt{y} \mapsto (\exists \mathtt{F} : f = (\mathtt{x} \mapsto \mathtt{F}(\mathtt{x})) \wedge \mathtt{y} = \mathtt{F}(x)))$

We can then show that

$$(\mathtt{x} \to F(\mathtt{x}))\text{`}a = F(a)$$

: $y = (\mathtt{x} \to F(\mathtt{x}))\text{`}a$ is equivalent to

$$(\exists \mathtt{G} : (\mathtt{x} \to F(\mathtt{x})) = (\mathtt{x} \mapsto \mathtt{F}(\mathtt{x})) \wedge y = \mathtt{G}(a)),$$

which by the axiom of courses of values is equivalent to

$$(\exists \mathtt{G} : (\forall \mathtt{x} : (F(\mathtt{x}) = \mathtt{G}(\mathtt{x})) \wedge y = \mathtt{G}(a)),$$

and this is equivalent to $y = F(a)$.

Frege then did a great deal of work before Russell disclosed the scandal.

Define the function $R$ as $\neg\mathbf{X}\text{`}\mathbf{X}$. Define $r = (\mathtt{x} \mapsto R(\mathtt{x}))$. Now $r\text{`}r = (\mathtt{x} \mapsto R(\mathtt{x}))\text{`}r = R(r) = \neg r\text{`}r$, which is a contradiction.

At this point Frege gave up.

We will argue that he shouldn't have. There is a rather subtle issue here. The mathematical warrant for our assertion is clear. We will describe a repair of his system which makes it consistent. This is a theorem. The question is whether this repair harmonizes with his philosophical aims. Our answer to this is, not perfectly, but not badly. There is a philosophical story which motivates the repair, but it has to be told carefully. It justifies Frege's mathematical work, but it casts doubt on his belief that what he was doing was pure logic.

Frege doesn't deny the existence of concepts of level higher than two (in fact, three: the second order universal quantifier is a third lvel function constant, for the same reason that the first order universal quantifier is a second order constant. If we provided a second level function placeholder $\mathcal{H}$ we could write this as $(\forall \mathtt{F} : \mathcal{H}(\mathtt{F}))$, but we prefer not even to provide this notational alternative. He regards the existence of courses of values as a

device to avoid talk of such concepts by pressing them down a level. He does say explicitly in his main work that there are concepts of all orders.

For example his account of the number 1 is as the course of values of a concept under which fall exactly those courses of values of objects under which just one object falls. In symbols, $1 = (\mathtt{x} \mapsto (\exists \mathtt{u} : (\forall \mathtt{v} : \mathtt{x}`\mathtt{v} = (\mathtt{u} = \mathtt{v})))$ But one could equally well think of the second level concept

$$(\exists \mathtt{u} : (\forall \mathtt{v} : \mathbf{F}(\mathtt{v}) = (\mathtt{u} = \mathtt{v})))).$$

This is the second level concept under which each first level concept falls which is true of only one thing. But this means that 1 is not an object. It is of a higher type. An account along these lines of arithmetic can be given. In fact, this is how Russell did it.

This means that numbers are not objects. It means that an account of equality of functions of the various levels has to be given (this is straightforward: $F = G$ is "of course" $(\forall \mathcal{P} : \mathcal{P}(F) = \mathcal{P}(G))$: note the immediate complication that we need quantifiers over second level functions. One could of course assert that this holds if $(\forall \mathtt{x} : F(\mathtt{x}) = G(\mathtt{x}))$ holds, but Frege knows that extensional criteria for identity of universals are fraught, and a beauty of his system is that he does not have to define equality for anything but objects. Now to assert that $x^2 - 5x + 6$ has two roots (an example of a kind Frege mentions) is to say that the concept under which 2 and 3 and no other second level functions fall falls under 2. But that would not be the same 2: this would be a fourth level concept under which falls third level concepts under which exactly two second level concepts fall. Russell gives a treatment just like this in his Principia. Frege despaired at the complexity of this.

Examination of what Frege is doing in his representation of number gives the clue as to how to get out of the predicament altogether. Acknowledge that we have functions and concepts of higher level in the back of our minds. When we write $(\mathtt{x} \mapsto F(\mathtt{x}))$, we are using an object to code a first level function. In an expression $f`x$, $f$ is an object playing the role of a function, and $x$ is being viewed as just an object. In the Russell concept $\neg x`x$, the same unknown $x$ appears in two different roles. This is what we propose to forbid.

We propose that each unknown $x$ will be assigned a natural number representing its role, and in this way the paradoxical function will become impossible to express. To every variable $u$, free or bound, object or function, we associate a natural number $\mathtt{level}(u)$. We will leave the levels tacit in our

notation, except that a place holder $\mathbf{X}$ or $\mathbf{F}$ must now be written $\mathtt{X}^i$ or $\mathtt{F}^{i+1}$: the level of a function must be at least 1. We provide an additional value $\infty$ for levels of expressions which are not of a fixed level.

We now indicate how to compute levels for all expressions, and how the computability of levels restricts the formation of functions and concepts.

To begin with, the level of any constant is $\infty$. This is not a type theory: every actual object we can specify is literally an object. So $\mathtt{level}(\mathbf{true}) = \mathtt{level}(\mathbf{false}) = \infty$.

The level of any proposition, even if unknown, will be $\infty$. This will require discussion.

We write $m \sim n$ for "$m = n$ or $m = \infty$ or $n = \infty$".

$\mathtt{level}(\mathtt{x} \mapsto F(\mathtt{x}))$ is well-formed iff $F(a)$ is well formed where $\mathtt{level}(a) = \mathtt{level}(\mathtt{x})$ and $\mathtt{level}(F(a)) \sim \mathtt{level}(a) = \mathtt{level}(\mathtt{x})$. The level of a course of values term is one level higher than that of its input variable, if it is not a constant: $\mathtt{level}(\mathtt{x} \mapsto F(\mathtt{x})) = \mathtt{level}(\mathtt{x}) + 1$ if there is an unknown in $F(x)$ other than $x$, and otherwise $\mathtt{level}(\mathtt{x} \mapsto F(\mathtt{x})) = \infty$.

The level of a variable is given. The level of $\mathbf{X}^i$ is $i$. The level of $\mathbf{F}^{i+1}$ is $i + 1$.

If $\mathtt{level}(t) \sim \mathtt{level}(u)$ ($t$ and $u$ possibly complex terms) then $t = u$ is an expression and $\mathtt{level}(t = u) = \infty$. Notice that this means that expressions that we equate must have the same level, or one of them must fail to have level.

For any terms $t, u$, we are given that $\neg t$ and $t \to u$ are terms with $\mathtt{level}(\neg t) = \mathtt{level}(t \to u) = \infty$.

For any term $F(x)$, and bound variable $\mathtt{x}$ of the same level as $x$ and not occurring in $F(x)$, we have $(\forall \mathtt{x} : F(\mathtt{x}))$ a term of level $\infty$.

For any function variable $F$ and term $t$ with $\mathtt{level}(t) \sim \mathtt{level}(F) - 1$, we have $F(t)$ a term of level $\mathtt{level}(F) - 1$.

For any term $\mathcal{P}(F)$ and variable $\mathtt{F}$ of the same type as $F$, $(\forall \mathtt{F} : \mathcal{P}(\mathtt{F}))$ is a term of level $\infty$.

For any term $t$ of level $i + 1$, $\backslash t$ is a term of level $i$. If $t$ is a term of level $\infty$, $\backslash t$ is a term of level $\infty$.

If $F(a)$ is an object term then $F(\mathbf{X}^{\mathtt{level}(a)})$ is a function term. It can be said to have level $\mathtt{level}(a) + 1$ if it contains unknowns other than $a$ and if the level of $F(a)$ is the same as the level of $a$. If it does not have level, that does not necessarily mean there is no use for it. Function terms which do not have level still may be used to define operations which change level: they may represent functions from level $i$ to level $i + k$ or vice versa, where $k$ is

the difference between the levels of $a$ and $F(a)$. So for example we have a function term $(\mathtt{x} \mapsto \mathtt{x} = \mathbf{X}^{\mathtt{level(x)}})$: this represents the operation $\iota$ taking an object $x$ to the course of values of the concept under which only $x$ stands: this is a reasonable candidate for the singleton set $\{x\}$, and it should be clear that it is not at the same level. $\iota$ doesn't represent a function that fits in our numbered sequence of types, but a lateral function from (some, or any) level $i$ to level $i+1$. The operation $\iota$ can be used in a well-formed definition, as long as when $t$ is of level $i$, $\iota(t)$ is understood to be of level $i+1$. There is no conflict with the scheme for function variables here, because $\iota$ is a constant.

If $F(a,b)$ then $F(\mathbf{X}^{\mathtt{level}(a)}, \mathbf{Y}^{\mathtt{level}(b)})$ is a term for an operation on two arguments. These may be used in definitions, but never to represent actual entities in our system. For example, the term

$$\backslash(\mathtt{y} \mapsto (\exists \mathtt{F} : \mathbf{X}^1 = (\mathtt{x} \mapsto \mathtt{F(x)}) \wedge \mathtt{y} = \mathtt{F}(\mathbf{Y}^0))$$

represents the course of values application operation (it can be read as defining $\mathbf{X}^1{`}\mathbf{Y}^0$: we could use any $\mathbf{X}^{i+1}$ and $\mathbf{Y}^i$, and from the levels of these parameters deduce the required levels of the other variables).

If $\mathcal{P}(F)$ is an object term then $\mathcal{P}(\mathbf{F}^{\mathtt{level}(F)})$ is a second level function term.

There is no question about whether this works mathematically. It is known that if ordinary set theory is consistent, the system described here is consistent. We will introduce one more assumption below to get Frege's arithmetic to work, which is also known to be consistent.

What needs to be asked is whether this makes sense philosophically. We are trying to restrict our language so that every unknown that we consider arises from varying an object when considered as representing entities of a particular level in a hierarchy in which level 0 is inhabited by objects and level $i+1$ is inhabited by functions from level $i$ objects to level $i$ objects. We fit concepts into this scheme by the device of assigning propositions indeterminate level. An unknown proposition $p$ is in itself a level 0 object. Note that $p$ is naively equivalent to $(\mathtt{x} \mapsto p \wedge x = x) = (\mathtt{x} \mapsto x = x)$, which would be assigned type 1 if we did not have special rules for propositions. This is an indication of how we could make this system work if we insisted on treating unknown propositions as having level; but we don't have to.

The Russell paradox goes away. $y = x{`}x$ expands to

$$\neg(\exists x = (\mathtt{u} \mapsto \mathtt{F(u)}) \wedge y = \mathtt{F}(x)).$$

Now the level of $\mathtt{F}$ must be one higher than the level of $x$, and the level of $y$ must be equal to the level of $\mathtt{F}(x)$, the same as the level of $x$. But also the level of $x$ must be the same as the level of $(\mathtt{u} : \mathtt{F}(\mathtt{u}))$, in which the level of $\mathtt{F}$ must be one higher than the level of $\mathtt{u}$, the level of $(\mathtt{u} \mapsto (\mathtt{F}(\mathtt{u})))$ must be one higher than that of $\mathtt{u}$ and so the same as that of $\mathtt{F}$, and finally the level of $\mathtt{x}$ must be the same as that of $(\mathtt{u} \mapsto (\mathtt{F}(\mathtt{u})))$ and so the same as that of $\mathtt{F}$, contradicting the requirements of the subterm $\mathtt{F}(x)$. The term $x`x$ is revealed to be ill-formed when its definition is expanded, and so the Russell function $\neg\mathbf{X}`\mathbf{X}$ cannot be formed. This is not a proof that this system works, but that it doesn't immediately fall into this pitfall. A non-paradoxical way of putting this is that if one looks at the definition of $f`x$, which does fit the level scheme, the level of $f$ is seen to be one higher than the level of $x$.

Now we can define 0 as $(\mathtt{x} \mapsto \mathtt{x} = (\mathtt{y} \mapsto \neg\mathtt{y} = \mathtt{y}))$, the course of values under which fall all courses of values of concepts under which zero objects fall.

We can define $\sigma$, the successor map (on numbers: it has eccentric effects on other concepts and functions) as

$$(\mathtt{n} \mapsto (\mathtt{z} \mapsto (\exists \mathtt{k} : \mathtt{n}`\mathtt{k} \wedge (\exists \mathtt{x} : \neg\mathtt{k}`\mathtt{x} \wedge (\forall \mathtt{u} : \mathtt{z}`\mathtt{u} = (\mathtt{k}`\mathtt{u} \vee \mathtt{u} = \mathtt{x})))))).$$

Wr provide a little assistance in reading this. Define $x \in y$ as

$$y`x \wedge (\forall \mathtt{z} : y`z = \neg\neg y`\mathtt{z}) :$$

$x \in y$ means that $y$ is the course of values of a concept under which $x$ falls. The level of $y$ has to be one higher than the level of $x$. Define $\{\mathtt{x} : P(\mathtt{x})\}$ as $(\mathtt{x} \mapsto \neg\neg P(\mathtt{x}))$. This has level one higher than the level of $x$ if it is not a closed term with level $\infty$. Define $\emptyset$ as $\{\mathtt{x} : \neg\mathtt{x} = \mathtt{x}\}$. Define $\{x\}$ as $\{\mathtt{y} : \mathtt{y} = x\}$. Notice that $\backslash\{x\} = x$. Define $x \cup y$ as $\{\mathtt{z} : \mathtt{z} \in x \vee \mathtt{z} \in y\}$. We feel free now to to refer to courses of values of concepts as sets and those objects which fall under the concept as elements of the set.

Then 0 has been defined as $\{\emptyset\}$. $\sigma(n)$ is defined as

$$\{z : (\exists \mathtt{k} : \mathtt{k} \in n \wedge (\exists \mathtt{x} : (\neg\mathtt{x} \in \mathtt{k}) \wedge \mathtt{z} = \mathtt{k} \cup \{\mathtt{x}\}))\}$$

The idea is that if $n$ is the set of all sets with $n$ elements, then $\sigma(n)$ will be the set of all sets with $n + 1$ elements.

Now define $\mathbb{N}$ as $\{\mathtt{n} : (\forall \mathtt{I} : (I(0) \wedge (\forall \mathtt{k} : I(\mathtt{k}) \to I(\sigma(\mathtt{k})))) \to I(n))\}$: the natural numbers are those objects which have all inductive properties.

This is the beginning of Frege's development of arithmetic. The entire development goes through, with two more significant remarks.

Frege wants to reify functions of two variables as objects. The way he did this was to reify a function $F(\mathbf{X}, \mathbf{Y})$ as $F^* = (\mathtt{y} \mapsto (\mathtt{x} \mapsto F(\mathtt{x}, \mathtt{y})))$. He calls this a double course of values. Then $(F^*`a)`b = F(a, b)$. Frege is the inventor of currying, it appears. This needs to be amended because it does not respect level. Instead, define $F^*$ as $(y \mapsto (x \mapsto F(\backslash x, y)))$ and observe that $(F^*`\{a\})`b = F(a, b)$ (where $F$ is a sensible function with two object arguments of the same level as its output).

The other revision is required to support Frege's proof that the universe is infinite. This relies on the assertion that $\{1, \ldots, n\}$ has $n$ elements. One would like to prove this by induction, but with arrangements so far it in fact does not follow. It can be made to follow if we define $\#n$ as

$$\backslash(\mathtt{m} \mapsto (\mathtt{m} \in \mathbb{N} \wedge \mathtt{m} = n) \vee (\neg \mathtt{m} \in \mathbb{N} \wedge \mathtt{m} = 0))$$

and provide that terms $\#n$ have level $\infty$. This has the effect of making numbers freely manipulable in level just as propositions are.

The mathematical basis for this repair of Frege's logic was laid by Quine in his definition of New Foundations in 1937. The repair was in effect completed by the proof of Jensen in 1969 that NFU, New Foundations weakened to enforce extensionality only for objects with elements is consistent and has $\omega$-models. An $\omega$-model of NFU satisfies Rosser's Axiom of Counting, $|\{1, \ldots, n\}| = n$, and the system just presented can be interpreted directly in NFU with Rosser's Axiom of Counting.

For convenience, we assume choice so that a type-level pair can be defined, following Quine. Then the universe of objects of our interpretation is simply the universe of sets of NFU + Counting. The True is interpreted as $V$ and the False as $\emptyset$. The functions of our interpretation are the functions with domain the universe in the usual sense of NFU + Counting (using the type level pair so that stratification aligns with our notion of level; it could be fixed if we used the usual Kuratowski pair, but it would add a layer of mystification). For each function $F$, $(\mathtt{x} \mapsto F(\mathtt{x}))$ is interpreted as...$F$. A function is simply the same as its course of values in our interpretation. It is then completely straightforward to check that all the axioms hold in this interpretation for reasons entirely familiar to workers in NFU. The Axiom of Counting ensures that variables restricted to natural numbers can be freely raised or lowered in type as required for applications of stratified comprehension; this is easy to show for variables restricted to the set of truth values.

There are three curious things to note here.

It is odd that functions and courses of values simply coincide.

It is noteworthy that the theory we present is not actually NFU + Rosser's Axiom of Counting, because we in fact cannot form unstratified sentences. It is known that all stratified sentences of NFU have stratified proofs; I suspect that it is true that all stratified sentences of NFU + Counting have proofs using stratification combined with the ability to raise and lower the types of natural number variables freely, but this should be checked. It is important to realize that the distinction between functions and objects here, though it looks formally like a distinction between second order quantification and first order quantification, is in fact not this in any real sense, since the domain of second order objects is exactly the same (actually a proper subset) of the domain of first order objects. We are not talking about MLU here.

This theory is surprisingly strong. NFU + Counting is known to have the strength of Zermelo set theory plus "$\beth_n$ exists" for each concrete natural number $n$. That is enough for all mathematics of Frege's day and almost all mathematics of our day.

I have an obligation to compare what I have done here with what Nino Cocchairella did: he has already pointed out that NFU can be used to implement a version of Frege's logic. I have a further obligation to point out that this development makes other "neo-Fregean" theories which are currently studied of questionable interest. These theories are extremely weak. The theory presented here is a fully capable foundation for mathematics, which is what Frege wanted.

We do not think that Frege, even as corrected by us (or Cocchiarella) , shows that mathematics reduces to logic. In fact, the repair (and the original paradox) bring out what the powerful non-logical assumption is. Postulating an injection from the domain of functions into the domain of objects is not a logical assumption but an assertion that the universe of objects is quite large. In the original version, it was large in a way that is impossible. In this version, it is large in a well-understood way, but a way the full understanding of which requires considerable logical and mathematical sophistication.