# Beginning to formalize a chapter on cardinals

We want to faithfully formalize the first lines of a chapter from a set theory scriptum in ForTheL so that it is accepted in the Naproche-SAD system. Since we concentrate on this chapter, earlier material can simply be „imported“ as signature extensions with axioms, instead of constructing that material and proving its properties.

## 1 Cardinalities

Apart from its foundational role, set theory is mainly concerned with the study of arbitrary infinite sets and in particular with the question of their size. Cantor's approach to infinite sizes follows naive intuitions familiar from finite sets of objects.

**Definition 1.**

*a)* *$x$ and $y$ are* equipollent*, or* equipotent*, or have* the same cardinality*, written $x \sim y$, if $\exists f f: x \leftrightarrow y$.*

*b)* *$x$ has* cardinality at most that of *$y$, written $x \preccurlyeq y$, if $\exists f f: x \rightarrow y$ is injective.*

*c)* *We write $x \prec y$ for $x \preccurlyeq y$ and $x \nsim y$.*

These relations are easily shown to satisfy

**Lemma 2.** *Assume* ZF*. Then*

*a)* *$\sim$ is an equivalence relation on V.*

*b)* $x \sim y \rightarrow x \preccurlyeq y \wedge y \preccurlyeq x$.

*c)* $x \preccurlyeq x$.

*d)* $x \preccurlyeq y \wedge y \preccurlyeq z \rightarrow x \preccurlyeq z$.

*e)* $x \subseteq y \rightarrow x \preccurlyeq y$.

… …

We begin like this:

```
# 8 Cardinalities

Definition 77a. x ~ y iff exists f f : x <-> y.
Let x and y are equipollent stand for x ~ y.
Let x and y have same cardinality stand for x ~ y.

Definition 77b. x \preccurlyeq y iff there exists f such that f : x ->
y and f is injective.

# Definition 77c.
Let x \preccurly y stand for x \preccurlyeq y and not x~ y.
```

This throws various errors:

- types of variables are not fixed, hence the variables are not accepted;
- relations like the ternary `f : x <-> y` are not defined.

To fix this we introduce appropriate language extensions (`Signature`):

```
# Preliminaries
Let x,y denote sets.
Let f denote functions.

Signature. f : x -> y is an atom.
Signature. f : x <-> y is an atom.
Signature. f is injective is an atom.

# 8 Cardinalities

Definition 77a. x ~ y iff exists f f : x <-> y.
Let x and y are equipollent stand for x ~ y.
Let x and y have same cardinality stand for x ~ y.

Definition 77b. x \preccurlyeq y iff there exists f such that f : x ->
y and f is injective.

# Definition 77c.
Let x \preccurly y stand for x \preccurlyeq y and not x~ y.
```

Notes:

- We use the inbuilt notions of `sets` and `functions`;
- the signature extensions provide notations for predicates that suggest their ordinary mathematical meanings; strictly speaking only a generic term with some typing has been introduced; `is an atom` means that we have introduced boolean valued terms, i.e., relations;
- one can use LaTeX-commands like `\preccurlyeq` for symbols; using some LaTeX-tools these can be typeset like $\preccurlyeq$;
- the symbolic patterns like `f : x <-> y` or `x and y are equipollent` can be chosen rather freely, but there are some restrictions to allow the proper parsing of various constructs: we cannot write `x and y have THE same cardinality` since the article `the` signals that some unique value or so is taken; so we leave out the `the`;
- there is a tradeoff between definitions and abbreviations. Minor definitions are sometimes better treated as abbreviations by the `Let ... stand for ...` construct. These abbreviations are already resolved by the parser and don't enter the first-order translation of texts.
- The first order translations can be seen by letting Naproche-SAD run with the `-T` option: `stack exec Naproche-SAD -- examples/regular_successor.ftl -T`

Now we want to prove the first property:

```
Lemma 78a1. x ~ x.
```

This is canonically proved using the identity function on $x$. We provide that function also in the preliminaries:

```
Signature. id x is a function such that id x : x \leftrightarrow x.
```

The automatic theorem prover (ATP) eprover is able to prove the Lemma using `id x`.

Here is a text which proves that $\sim$ is an equivalence relation, corresponding to the original scriptum.

```
# Preliminaries

Let x,y,z stand for sets.

Let f,g stand for functions.

Signature. f : x \rightarrow y is an atom.
Signature. f : x \leftrightarrow y is an atom.
Signature. f is injective is an atom.

Signature. id x is a function such that id x : x \leftrightarrow x.

Signature. Assume f : x \leftrightarrow y.
inv(f,x,y) is a function such that inv(f,x,y) : y \leftrightarrow x.

Signature.
Assume f : x \leftrightarrow y. Assume g : y \leftrightarrow z.
comp(g,f,x,y,z) is a function such that comp(g,f,x,y,z) : x
\leftrightarrow z.

# 8 Cardinalities

Definition 77a. x \sim y iff exists f f : x \leftrightarrow y.
Let x and y are equipollent stand for x \sim y.
Let x and y are equipotent stand for x \sim y.
Let x and y have same cardinality stand for x \sim y. # "the same" not
accepted.
Let x \nsim y stand for not x \sim y.
Definition 77b. x \preccurlyeq y iff
exists f (f : x \rightarrow y and f is injective).
# Definition 77c.
Let x \preccurly y stand for x \preccurlyeq y and x \nsim y.

Lemma 78a1.
x \sim x.
Lemma 78a2.
If x \sim y then y \sim x.
Lemma 78a3.
If x \sim y and y \sim z then x \sim z.
```

Note:

- The proofs of the Lemmas are based on the existence of certain (abstract) functions through the signature extensions; eprover finds these proofs without further help;
- the preliminaries are intended to describe aspects of a standard model of sets and functions; if one would leave out some preconditions of the signature extensions, the axioms get even stronger, but they may no longer correspond to standard models.

Conclusions:

- The preliminaries contain abstract operations on functions like inversion and composition which one would find in a category of isomorphisms;

- In a more comprehensive set theoretical approach, the preliminaries would be the result of a theory of functions;

- Although ForTheL is part of natural language, the requirement for grammatical acceptance and logic correctness are much more strict and subtle than in ordinary (mathematical) language since there is no intelligent reader who is able to complete incomplete specifications.