The **numspell** package

v1.0 (2017/02/09)

Tibor Tómács
tomacs.tibor@uni-eszterhazy.hu

1 Introduction

The aim of the `numspell` package is to spell the cardinal and ordinal numbers from 0 to $10^{67} - 1$ (i.e. maximum 66 digits).

Currently, the supported languages are English, German, French and Hungarian. The spelling will happen in the current language.

The `numspell` package requires the services of the following packages: `xstring`, `etoolbox`, `pdftexcmds`.

Load the package as usual, with

```
\usepackage{numspell}
```

2 Commands

`\numspell[`⟨*zeros*⟩`]{`⟨*num*⟩`}`

Spelling the cardinal number $n = \langle num \rangle \cdot 10^{\langle zeros \rangle}$, where $0 \leq n \leq 10^{67} - 1$. The default value of ⟨*zeros*⟩ is `0`. For example

`\numspell{12000}` → twelve thousand
`\numspell[3]{12}` → twelve thousand
`\numspell[6]{12}` → twelve million
`\numspell[63]{1}` → one vigintillion

`\thenumspell`

The `\numspell` stores the result in this command. For example

`\numspell{12000}; \thenumspell` → twelve thousand; twelve thousand
`\numspell{1}; \numspell{2}; \thenumspell` → one; two; two

`\numspellsave{`⟨*name*⟩`}`

It generates the `\thenumspell`⟨*name*⟩ command, which saves the current `\thenumspell`. For example

```
\numspell{1};
\numspellsave{MyNum}
\numspell{2};
\thenumspell;
\thenumspellMyNum
```

one; two; two; one

`\numspelldashspace{`⟨*length*⟩`}`

In the number spelling, the spaces around the dashes are flexibility for the optimal hyphenation. Its value is `0pt plus` ⟨*length*⟩. The default value of ⟨*length*⟩ is `2pt`. For example

```
\selectlanguage{magyar}
\numspell{651231235476254716254625 4756}\\[2mm]
\numspelldashspace{10pt}
\numspell{651231235476254716254625 4756}
```

hatkvadrilliárd-ötszáztizenkétkvadrillió-háromszáztizenkéttrilliárd-háromszázötvennégytrillió-hétszázhatvankétbilliárd-ötszáznegyvenhétbillió-egyszázhatvankétmilliárd-ötszáznegyvenhatmillió-kétszázötvennégyezer-hétszázötvenhat

hatkvadrilliárd - ötszáztizenkétkvadrillió - háromszáztizenkéttrilliárd - háromszázötvennégytrillió - hétszázhatvankétbilliárd-ötszáznegyvenhétbillió-egyszázhatvankétmilliárd-ötszáznegyvenhatmillió-kétszázötvennégyezer-hétszázötvenhat

`\numspell*[`⟨*zeros*⟩`]{`⟨*num*⟩`}`

It works like `\numspell`, but the number spelling will not be printed. In other words, the following two lines are equivalent:

```
\numspell[⟨zeros⟩]{⟨num⟩}
\numspell*[⟨zeros⟩]{⟨num⟩}\thenumspell
```

For example

```
\numspell*{1}
\numspellsave{MyNum}
\numspell*{2}
\thenumspell;
\thenumspellMyNum
```

two; one

`\Numspell[`⟨*zeros*⟩`]{`⟨*num*⟩`}`

It works like `\numspell`, but the first letter will be capital. For example

`\Numspell{12000}` → Twelve thousand
`\Numspell[3]{12}` → Twelve thousand
`\Numspell[6]{12}` → Twelve million
`\Numspell[63]{1}` → One vigintillion

`\Numspell*[`⟨*zeros*⟩`]{`⟨*num*⟩`}`

It works like `\Numspell`, but the number spelling will not be printed. In other words, the following two lines are equivalent:

```
\Numspell[⟨zeros⟩]{⟨num⟩}
\Numspell*[⟨zeros⟩]{⟨num⟩}\thenumspell
```

For example

```
\Numspell*{1}
\numspellsave{MyNum}
\Numspell*{2}
\thenumspell;
\thenumspellMyNum
```

Two; One

`\ordnumspell[`⟨*zeros*⟩`]{`⟨*num*⟩`}`

Spelling the ordinal number $n = \langle num \rangle \cdot 10^{\langle zeros \rangle}$, where $0 \leq n \leq 10^{67} - 1$. The default value of ⟨*zeros*⟩ is 0. For example

`\ordnumspell{12000}` → twelve thousandth
`\ordnumspell[3]{12}` → twelve thousandth
`\ordnumspell[6]{12}` → twelve millionth
`\ordnumspell[63]{1}` → one vigintillionth

`\ordnumspell*[⟨zeros⟩]{⟨num⟩}`
It works like `\ordnumspell`, but the number spelling will not be printed. In other words, the following two lines are equivalent:

```
\ordnumspell[⟨zeros⟩]{⟨num⟩}
\ordnumspell*[⟨zeros⟩]{⟨num⟩}\thenumspell
```

For example

```
\ordnumspell*{1}
\numspellsave{MyNum}
\ordnumspell*{2}
\thenumspell;
\thenumspellMyNum
```

second; first

`\Ordnumspell[⟨zeros⟩]{⟨num⟩}`
It works like `\ordnumspell`, but the first letter will be capital. For example

`\Ordnumspell{12000}` → Twelve thousandth
`\Ordnumspell[3]{12}` → Twelve thousandth
`\Ordnumspell[6]{12}` → Twelve millionth
`\Ordnumspell[63]{1}` → One vigintillionth

`\Ordnumspell*[⟨zeros⟩]{⟨num⟩}`
It works like `\Ordnumspell`, but the number spelling will not be printed. In other words, the following two lines are equivalent:

```
\Ordnumspell[⟨zeros⟩]{⟨num⟩}
\Ordnumspell*[⟨zeros⟩]{⟨num⟩}\thenumspell
```

For example

```
\Ordnumspell*{1}
\numspellsave{MyNum}
\Ordnumspell*{2}
\thenumspell;
\thenumspellMyNum
```

Second; First

3 Commands for English language

`\numspellUS`
By default, the number spelling will happen in British English, if the `english` language is active. This command changes it to American English. For example

`\numspellUS\numspell{1012345}` → one million, twelve thousand, three hundred forty-five

`\numspellGB`
Using the `\numspellUS` command, you can rechange it to British English by this command. For example

```
\numspellUS\numspell{1012345}\\
\numspellGB\numspell{1012345}
```

one million, twelve thousand, three hundred forty-five
one million, twelve thousand and three hundred and forty-five

4 Commands for French language

The following commands only work, if `french` language is active.

`\numspellpremiere`
By default, `\ordnumspell{1}` → premier,
but `\numspellpremiere\ordnumspell{1}` → première

`\numspellpremier` (default)
`\numspellpremiere\ordnumspell{1};`
`\numspellpremier\ordnumspell{1}`

première ; premier

5 Commands for Hungarian language

The following commands only work, if `magyar` language is active.

`\anumspell[`⟨*zeros*⟩`]{`⟨*num*⟩`}`
It works like `\numspell`, but the number spelling will start with Hungarian definite article. For example

`\anumspell{1}` → az egy
`\anumspell{2}` → a kettő

`\anumspell*[`⟨*zeros*⟩`]{`⟨*num*⟩`}`
It works like `\anumspell`, but the number spelling will not be printed. In other words, the following two lines are equivalent:

```
\anumspell[⟨zeros⟩]{⟨num⟩}
\anumspell*[⟨zeros⟩]{⟨num⟩}\thenumspell
```

For example

```
\anumspell*{1}
\numspellsave{MyNum}
\anumspell*{2}
\thenumspell;
\thenumspellMyNum
```

a kettő; az egy

`\Anumspell[`⟨*zeros*⟩`]{`⟨*num*⟩`}`
It works like `\anumspell`, but the first letter will be capital. For example

`\Anumspell{1}` → Az egy
`\Anumspell{2}` → A kettő

`\Anumspell*[`⟨*zeros*⟩`]{`⟨*num*⟩`}`
It works like `\Anumspell`, but the number spelling will not be printed. In other words, the following two lines are equivalent:

```
\Anumspell[⟨zeros⟩]{⟨num⟩}
\Anumspell*[⟨zeros⟩]{⟨num⟩}\thenumspell
```

For example

```
\Anumspell*{1}
\numspellsave{MyNum}
\Anumspell*{2}
\thenumspell;
\thenumspellMyNum
```

A kettő; Az egy

\aordnumspell[⟨*zeros*⟩]{⟨*num*⟩}

It works like `\ordnumspell`, but the number spelling will start with Hungarian definite article. For example

```
\aordnumspell{1} → az első
\aordnumspell{2} → a második
```

\aordnumspell*[⟨*zeros*⟩]{⟨*num*⟩}

It works like `\aordnumspell`, but the number spelling will not be printed. In other words, the following two lines are equivalent:

```
\aordnumspell[⟨zeros⟩]{⟨num⟩}
\aordnumspell*[⟨zeros⟩]{⟨num⟩}\thenumspell
```

For example

```
\aordnumspell*{1}
\numspellsave{MyNum}
\aordnumspell*{2}
\thenumspell;
\thenumspellMyNum
```

a második; az első

\Aordnumspell[⟨*zeros*⟩]{⟨*num*⟩}

It works like `\aordnumspell`, but the first letter will be capital. For example

```
\Aordnumspell{1} → Az első
\Aordnumspell{2} → A második
```

\Aordnumspell*[⟨*zeros*⟩]{⟨*num*⟩}

It works like `\Aordnumspell`, but the number spelling will not be printed. In other words, the following two lines are equivalent:

```
\Aordnumspell[⟨zeros⟩]{⟨num⟩}
\Aordnumspell*[⟨zeros⟩]{⟨num⟩}\thenumspell
```

For example

```
\Aordnumspell*{1}
\numspellsave{MyNum}
\Aordnumspell*{2}
\thenumspell;
\thenumspellMyNum
```

A második; Az első

6 Examples

Example 1

```
\documentclass{article}
\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[magyar,german,french,english]{babel}
\usepackage{numspell}
\usepackage[group-separator={,}]{siunitx}
\begin{document}

\def\mynum{123456789}

\noindent
In British English the spelling of \num{\mynum} is
\emph{``\numspell{\mynum}''}.

\smallskip\noindent
In American English the spelling of \num{\mynum} is
{\numspellUS\emph{``\numspell{\mynum}''}}.

\smallskip\noindent
In German the spelling of \num{\mynum} is
{\selectlanguage{german}\emph{``\numspell{\mynum}''}}.

\smallskip\noindent
In French the spelling of \num{\mynum} is
{\selectlanguage{french}\emph{``\numspell{\mynum}''}}.

\smallskip\noindent
In Hungarian the spelling of \num{\mynum} is
{\selectlanguage{magyar}\emph{``\numspell{\mynum}''}}.

\end{document}
```

In British English the spelling of 123,456,789 is *“one hundred and twenty-three million, four hundred and fifty-six thousand and seven hundred and eighty-nine”*.

In American English the spelling of 123,456,789 is *“one hundred twenty-three million, four hundred fifty-six thousand, seven hundred eighty-nine”*.

In German the spelling of 123,456,789 is *“einhundertdreiundzwanzig Millionen vierhundertsechsundfünfzigtausendsiebenhundertneunundachtzig”*.

In French the spelling of 123,456,789 is *“cent vingt-trois millions quatre cent cinquante-six mille sept cent quatre-vingt-neuf”*.

In Hungarian the spelling of 123,456,789 is *“százhuszonhárommillió-négyszázötvenhatezer-hétszáznyolcvankilenc”*.

Example 2

```
\documentclass{article}
\usepackage{numspell}
\usepackage[group-separator={,}]{siunitx}
\begin{document}
```

```
\def\mynum{123456789012345678901234567890123456789012345678901234567890123456}
\Numspell{\mynum}, that is \num{\mynum}.

\end{document}
```

One hundred and twenty-three vigintillion, four hundred and fifty-six novemdecillion, seven hundred and eighty-nine octodecillion, twelve septendecillion, three hundred and forty-five sexdecillion, six hundred and seventy-eight quindecillion, nine hundred and one quattuordecillion, two hundred and thirty-four tredecillion, five hundred and sixty-seven duodecillion, eight hundred and ninety undecillion, one hundred and twenty-three decillion, four hundred and fifty-six nonillion, seven hundred and eighty-nine octillion, twelve septillion, three hundred and forty-five sextillion, six hundred and seventy-eight quintillion, nine hundred and one quadrillion, two hundred and thirty-four trillion, five hundred and sixty-seven billion, eight hundred and ninety million, one hundred and twenty-three thousand and four hundred and fifty-six, that is 123,456,789,012,345,678,901,234,567,890,123,456,789,012,345,678,901,234,567,890,123,456.

Example 3

```
\documentclass{article}
\usepackage{numspell}
\newcounter{mycount}
\makeatletter
\begin{document}

The
\@whilenum\value{mycount}<31
\do{\ordnumspell{\themycount}\stepcounter{mycount},\ }\dots

\end{document}
```

The zeroth, first, second, third, fourth, fifth, sixth, seventh, eighth, ninth, tenth, eleventh, twelfth, thirteenth, fourteenth, fifteenth, sixteenth, seventeenth, eighteenth, nineteenth, twentieth, twenty-first, twenty-second, twenty-third, twenty-fourth, twenty-fifth, twenty-sixth, twenty-seventh, twenty-eighth, twenty-ninth, thirtieth, ...

Example 4

```
\documentclass{article}
\usepackage{numspell}
\newcounter{mycount}
\def\themycount{\numspell{\arabic{mycount}}}
\makeatletter
\begin{document}

\Numspell{0},
\@whilenum\value{mycount}<30
\do{\stepcounter{mycount}\themycount,\ }\dots

\end{document}
```

Nought, one, two, three, four, five, six, seven, eight, nine, ten, eleven, twelve, thirteen, fourteen, fifteen, sixteen, seventeen, eighteen, nineteen, twenty, twenty-one, twenty-two, twenty-three, twenty-four, twenty-five, twenty-six, twenty-seven, twenty-eight, twenty-nine, thirty, ...

7 Limitations

Do not use the `\numspell`, `\numspell*`, `\Numspell`, `\Numspell*`, etc. commands inside `\MakeUppercase` and sectioning commands. An example for the illustration of the problem:

```
\documentclass{article}
\usepackage{hyperref,numspell}
\pagestyle{headings}
\begin{document}

\section{The \ordnumspell{123} factor}
\MakeUppercase{\numspell{123}}
\newpage
Text

\end{document}
```

The bugs:

1. You can see it on the page 1: “one hundred and twenty-three”
   Required: “ONE HUNDRED AND TWENTY-THREE”
2. You can see it on the heading: *“THE one hundred and twenty-third FACTOR”*
   Required: *“THE ONE HUNDRED AND TWENTY-THIRD FACTOR”*
3. You can see it on the pdf bookmark: “The 123 factor”
   Required: “The one hundred and twenty-third factor”

The solution is very easy:

```
\ordnumspell*{123}
\section{The \thenumspell\ factor}
\numspell*{123}
\MakeUppercase{\thenumspell}
```