moreenum

Seamus Bradley

November 3, 2011

This package provides more enumeration styles for enumerate environments. The styles are supposed to work with enumitem. This is moreenum version 1.03.

1 Basic procedure

At the heart of each new enumeration is the following procedure:

```
\newcommand*{\macro}[1]{%
  \expandafter\@macro\csname c@#1\endcsname}
\newcommand*{\@macro}[1]{%
  \translate{#1}}
\AddEnumerateCounter{\macro}{\@macro}{distance}
```

From a user perspective, \macro takes a counter as its argument and outputs, say, a binary number or whatever you want. Actually, what it really does is turn a counter into a number and pass the number to \@macro which does the real work. It takes a number and translates it into the final representation.

Most of the cleverness is done by \translate and these are mostly macros I've borrowed from other packages.

The distance is the widest entry in the enumeration. moreenum hasn't been tested much with this parameter: I've just guessed a bit at what's the widest enumerations are likely to get. Enumerations can *theoretically* get up to 2147483647 items long. Which would be rather a long number.[1]

The \greek macro is a little more involved because it involves first defining a macro that turns numbers into Greek letters.

```
\newcommand*{\single@greek}[1]{%
  \expandafter\@single@greek\csname c@#1\endcsname
}
```

[1] fmcount doesn't seem to work with numbers that big, actually. But even 131071 is 11111111111111111

```
\newcommand*{\@single@greek}[1]{%
  $\ifcase#1\or\alpha\or\beta\or\gamma\or\delta\or\varepsilon
    \or\zeta\or\eta\or\theta\or\iota\or\kappa\or\lambda
    \or\mu\or\nu\or\xi\or o\or\pi\or\varrho\or\sigma
    \or\tau\or\upsilon\or\phi\or\chi\or\psi\or\omega
    \else\@ctrerr\fi$
}
```

Then you need to define what to do when you run out of letters. You start again at $\alpha\alpha$. The clever work there is done by the `alphalph` package.

```
\newalphalph{\@greek}[alph]{\@single@greek}{24}
\newcommand*{\greek}[1]{%
  \expandafter\@greek\csname c@#1\endcsname
}

\AddEnumerateCounter{\greek}{\@greek}{$\omega$}
```

Some sophistication is required to get the `\translate`-style macros to play nice with `\label` and `\ref` facilities. This can be seen in the following example.

```
\newcommand*{\enumHex}[1]{%
  \expandafter\@enumHex\csname c@#1\endcsname}
\newcommand*{\@enumHex}[1]{%
  \protect\Hexadecimalnum{\number#1}}
\AddEnumerateCounter{\enumHex}{\@enumHex}{AAAA}
```

The `\protect` makes sure the `\Hexadecimalnum` get written to the `.aux` file, rather than expanded first. The `\number` makes sure the number *is* written to the `.aux` file.[2]

2 Limitations

The biggest number TeX can handle is 2147483647. I can't imagine this ever being a serious limitation to your enumerating.

There are, however, some further limitations. Certain `fmtcount` macros seem to break before they hit this fundamental limit. In brackets are the `moreenum`-defined enumerations affected.

- `\binary` and friends break at 131072 [`\enumbinary`]
- `\hexadecimal` and friends break at 1048576 [`\enumhex` and `\enumHex`]
- `\numberstring` and friends break at 100000 [`\nwords`, `\nthwords` and friends]

[2] I'm actually guessing here. I have no idea. I got the clue from egreg here: `http://tex.stackexchange.com/q/22234/215`

None of these is a serious limitation. If you desperately need bigger enumerations, they are fairly straightforward to define yourself using `binhex` for the numbers and `numname` for the words: these packages don't have these limitations.[3]

3 Examples of the enumerations

Here are examples of all the kinds of enumeration that the package defines. The first item contains a reference to the third. This is to test if the referencing is working. The labels have dots after them, to check whether errant spaces are being added after the labels.[4]

`\greek`

α. Liberty: γ.

β. Equality

γ. Fraternity

$\alpha\sigma$. Meaning of life

`\Greek`

A. Liberty: Γ.

B. Equality

Γ. Fraternity

$A\Sigma$. Meaning of life

`\enumHex`

1. Liberty: 3.
2. Equality
3. Fraternity

2A. Meaning of life

`\enumhex`

1. Liberty: 3.
2. Equality
3. Fraternity

2a. Meaning of life

`\enumbinary`

1. Liberty: 11.

10. Equality

11. Fraternity

101010. Meaning of life

`\enumoctal`

1. Liberty: 3.
2. Equality
3. Fraternity

52. Meaning of life

[3] Why don't I just use those packages instead? Because having `fmtcount` do most of the work means only loading one package instead of 3 (`numname`, `binhex` and `nth` or `engord`). Also, `fmtcount` can speak different languages, and in future releases I'm tempted to try to get that working here.

[4] Thanks to Kevin Klement for pointing this issue out to me.

\raisenth

1st. Liberty: 3rd.

2nd. Equality

3rd. Fraternity

42nd. Meaning of life

\levelnth

1st. Liberty: 3rd.

2nd. Equality

3rd. Fraternity

42nd. Meaning of life

\Nthwords

First. Liberty: Third.

Second. Equality

Third. Fraternity

Forty-Second. Meaning of life

\Nwords

One. Liberty: Three.

Two. Equality

Three. Fraternity

Forty-Two. Meaning of life

\NTHWORDS

FIRST. Liberty: THIRD.

SECOND. Equality

THIRD. Fraternity

FORTY-SECOND. Meaning of life

\NWORDS

ONE. Liberty: THREE.

TWO. Equality

THREE. Fraternity

FORTY-TWO. Meaning of life

\nthwords

first. Liberty: third.

second. Equality

third. Fraternity

forty-second. Meaning of life

\nwords

one. Liberty: three.

two. Equality

three. Fraternity

forty-two. Meaning of life