The `longdivision` package defines two main commands: `\longdivision` and `\intlongdivision`. The usage for both is `\longdivision[⟨options⟩]{⟨dividend⟩}{⟨divisor⟩}`. The difference is that `\longdivision` divides until the remainders repeat or the quotient has too many digits to fit the page, whereas `\intlongdivision` does integer division and leaves the remainder. The command `\longdivisionkeys{⟨options⟩}` is also defined to set default options. At most 20 division steps worth of work will be displayed and at most 60 digits worth of division output will be produced. Thanks to Mike Jenck, Cameron McLeman, and Yu-Tsung Tai for emailing me with bug reports and feature requests.

Here is an example usage:

\begin{verbatim}
\longdivision{100}{22} \quad \intlongdivision{100}{22}
\end{verbatim}

\begin{verbatim}
4.54
22 \overline{100.00}
88
72.0
11.0
1.00
88
72
\end{verbatim}

These commands have several key-value options:

**max extra digits = ⟨nonnegative integer⟩**

This key determines the maximum amount of “extra” zeroes to add to the end of the dividend in the process of division – if the quotient has more digits before it repeats, the division will just stop. This is only an option for `\longdivision`, the command `\intlongdivision{⟨dividend⟩}{⟨divisor⟩}` is equivalent to `\longdivision[max extra digits=0]{⟨dividend⟩}{⟨divisor⟩}`. For brevity, this option has the short form where just the value is provided: `\longdivision[2]{10.0}{7}` is the same as `\longdivision[max extra digits=2]{10.0}{7}`.

\begin{verbatim}
\longdivision[max extra digits = 2]{10.0}{7} \quad \intlongdivision{100}{22}
\end{verbatim}

\begin{verbatim}
\longdivision[stage = 2]{10.0}{7} \quad \intlongdivision{100}{22}
\end{verbatim}

**stage = ⟨nonnegative integer⟩**

This controls how many steps worth of division to do. Thanks to Cam McLeman for suggesting this feature.
\longdivision{1}{7} \quad \longdivision{1}{7} \quad \longdivision{1}{7} \quad \longdivision{1}{7} \quad \longdivision{1}{7} \quad \longdivision{1}{7}

<table>
<thead>
<tr>
<th>7</th>
<th>1</th>
<th>0.1</th>
<th>0.14</th>
<th>0.142</th>
<th>0.1428</th>
<th>0.14285</th>
<th>0.142857</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>28</td>
<td>28</td>
<td>28</td>
<td>28</td>
<td>28</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>56</td>
<td>56</td>
<td>56</td>
<td>56</td>
<td>56</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

You can define your own typesetting style by saying \longdiv{\textit{style} = \{\textit{name}\} \{\textit{code}\}}. This can use the commands \longdiv{\textit{dividend}} which contains the dividend, \longdiv{\textit{divisor}} which contains the divisor, \longdiv{\textit{quotient}} which contains the quotient, and \longdiv{\textit{work}} which contains the division work. For instance, a simplified version of the \textit{german} style is:

\longdivision{100.0}{13} \quad \longdivision{100.0}{13} \quad \intlongdivision{100.0}{13} \quad \intlongdivision{100.0}{13}

\textbf{style = \{style\}}

Control the style for typesetting the result of long division. The options are \texttt{default}, \texttt{standard}, \texttt{tikz}, or \texttt{german}. The option \texttt{default} is the same as \texttt{tikz} if Ti\textsc{k}Z is loaded and otherwise is the same as \texttt{standard}. You probably should load Ti\textsc{k}Z because the Ti\textsc{k}Z version looks significantly better. If you use this option, you'll probably want to set the style once and for all in your preamble with \longdivisionkeys{\textit{style} = \{\textit{style}\}}.
\longdivdefinestyle{my style}{%
begin{tabular}{@{}}
\longdivdividend \, , \longdivdivisor \, = \longdivquotient \\
\longdivwork
\end{tabular}
}%
\longdivision[style = my style]{2}{3}

2.0: \(3 = 0.\overline{5}\)
1.8
\[
2
\]

Send me an email if you cannot figure out how to make a style to your liking.

repeating decimal style = ⟨style⟩

Control the way that repeating decimals are typeset. The options are overline, dots, dots all, parentheses, or none. The default is overline. The parentheses style creates ugly spacing problems and the dots style is insufficiently visible, so the overline style is the best. If you use this option, you’ll probably want to set the style once and for all in your preamble with \longdivisionkeys{recurring decimal style=⟨style⟩} \longdivisionkeys{style=⟨style⟩}. Like the style key, this is designed to be extensible. However, the process of creating new repeating decimal styles is a bit involved. Send me an email if you want a new repeating decimal style.

\longdivision{repeating decimal style = overline }{5.3}{37} \quad
\longdivision{repeating decimal style = dots }{5.3}{37} \quad
\longdivision{repeating decimal style = dots all }{5.3}{37} \quad
\longdivision{repeating decimal style = parentheses}{5.3}{37} \quad
\longdivision{repeating decimal style = none }{5.3}{37}

\begin{tabular}{ccc}
\hline
0.1\overline{432} & 0.1\overline{432} & 0.1\overline{432} & 0.1\overline{(432)} \\
37 \hspace{1cm} 5.3000 \hspace{1cm} 37 \hspace{1cm} 5.3000 \hspace{1cm} 37 \hspace{1cm} 5.3000 \\
\hline
3.7 & 3.7 & 3.7 & 3.7 \\
1.60 & 1.60 & 1.60 & 1.60 \\
1.48 & 1.48 & 1.48 & 1.48 \\
120 & 120 & 120 & 120 \\
111 & 111 & 111 & 111 \\
90 & 90 & 90 & 90 \\
74 & 74 & 74 & 74 \\
16 & 16 & 16 & 16 \\
\hline
\end{tabular}