

The sillypage package^{*}

Phelype *lemon curry* Oleinik[†]
Paulo *dictates a letter* Cereda[‡]
samcarter *cheesy comestibles*[§]
Ulrike *flying lessons* Fischer[¶]

Released 2022-02-02

1 Introduction

This package is inspired by *The Ministry of Silly Walks*, a famous sketch from Monty Python. According to the corresponding [Wikipedia article](#):

The sketch as originally depicted in the series begins with John Cleese playing Mr. Teabag, a civil servant who, after purchasing The Times from the newsagent in the previous sketch, walks through the streets of London (at the crossing of Thorpebank Road and Dunraven Road) in a very peculiar manner. He eventually arrives at his place of business: The Ministry of Silly Walks, on the northern end of Whitehall. In the hallway, he passes other employees all exhibiting their own silly walks before arriving at his office. Once there, he finds Mr. Putey (Michael Palin) waiting for him and apologizes for the delay, explaining that his walk has become particularly silly of late and it takes longer for him to reach his destination.

So we decided to bring this masterpiece of British pop culture to the \TeX world through a silly page numbering style. Make sure to use this package in your thesis, it will be a success.

2 Usage

Using a silly package like this is quite straightforward! For starters, make sure to include it in your document preamble through

```
\usepackage{sillypage}
```

^{*}This file has version number 1.4, last revised 2022-02-02.

[†]E-mail: [phelype.oleinik\(at\)latex-project.org](mailto:phelype.oleinik@latex-project.org)

[‡]E-mail: [cereda.paulo\(at\)gmail.com](mailto:cereda.paulo@gmail.com)

[§]E-mail: [sam.carter8\(at\)aol.com](mailto:sam.carter8@aol.com)

[¶]E-mail: [fischer\(at\)troubleshooting-tex.de](mailto:fischer@troubleshooting-tex.de)



and that is it! The available interface is as follows:

`\pagenumbering{silly}`

`\pagenumbering{silly}`

Write `\pagenumbering{silly}` in you document to use the page numbering style defined by this package. According to Wikipedia, the typical silly walk gait consists of twelve specific steps to be performed, so page numbering will cycle through these steps, in an kineograph fashion.

`\silly`

`\silly {<counter>}`

Use the macro `\silly` on a ~~TeX~~ `<counter>` to typeset the corresponding image for the value of that counter. As noted above, the value will be mapped to an image in the silly walk cycle. Also, observe that this macro is applied to counters and not to integers.

`\sillystep`

`\sillystep {<integer>}`

Prints the step number `<integer>` from the sequence of steps. This macro works exactly like `\silly`, but on integers instead of counters. The integer value will be mapped to an image in the silly walk cycle.

`\pagenumbering{sillynumeral}`

`\pagenumbering{sillynumeral}`

Write `\pagenumbering{sillynumeral}` in you document to use a silly numeral system, in which each page will be converted to a unique composition of silly steps. Note that this macro differs from its `\silly` counterpart, as the former is a proper base 12 numeral system whereas the latter simply walks through a 12-cycle silly routine.

`\sillynumeral`

`\sillynumeral {<integer>}`

Provides the silly numeral system representation from the provided `<integer>` value. This is a proper base 12 numeral system.

What a silly yet marvellous interface!

3 A sample document

As a means to demonstrate how straightforward (and of course silly) our package is, here is a minimal working example:

```
\documentclass{article}
```

```
\usepackage{sillypage}
```

```
\usepackage{blindtext}
```

```
\begin{document}
```

```
\pagenumbering{silly}
```

```
\Blinddocument
```

```
\end{document}
```



After compiling it, take a look at the page numbering: you will be greeted with the silly walk routine! To impress your supervisor, use our package in your thesis, print it and ask them to view those pages in quick succession, so the images appear to animate by simulating motion! It is known to work with thesis committees as well.

4 The silly walk routine

This is the complete silly walk routine as described by Jazeen Hollings in the [corresponding Wikipedia article](#). Please note that the images used in this package are created by samcarter and released under the [CC BY-SA 4.0](#) license. Image sources (vector format) are available in the [package repository](#)



We also recommend Pilates.

5 Implementation details

The implementation is quite straightforward. Feel free to contact us for suggestions and criticism, code improvements and of course other silly walk variants.

```
1 <*package>
2 <@@=silly>
```

5.1 Package requirements

```
3 \ProvidesExplPackage
4   {sillypage} {sillypageDate} {sillypageVersion}
5   {The silly walk as page numbering style}
6 \RequirePackage{graphicx}
```

5.2 Picture-specific macros

```
\silly@steps
\silly@picture
\silly@height
7 \newcommand*\silly@steps{12}
8 \newcommand*\silly@height{.8cm}
9 \newcommand*\silly@picture{sillywalk-map.pdf}

(End definition for \silly@steps, \silly@picture, and \silly@height.)
```

5.3 Single step macros


```
\silly
\sillystep
10 \NewDocumentCommand \silly { m } { \exp_args:Nv \__silly_step:n { c@#1 } }
11 \NewDocumentCommand \sillystep { m } { \__silly_step:n { #1 } }
12 \newcommand*\@silly { \__silly_step:n }
13 \cs_new:Npn \__silly_step:n #1
14 {
```



```

15     \__silly_step_aux:eN
16     { \int_eval:n { \int_mod:nn { #1-1 } { \silly@steps } +1 } }
17     \c_false_bool
18 }

```


(End definition for \silly and others. These functions are documented on page  .)

5.4 Multi step macros

```

\sillynumeral
\@sillynumeral
\__silly_numeral:n
\__silly_numeral_aux:n
19 \NewDocumentCommand \sillynumeral { m } { \__silly_numeral:n {#1} }
20 \newcommand*\@sillynumeral { \__silly_numeral:n }
21 \cs_new:Npn \__silly_numeral:n #1
22 {
23     \exp_args:Ne \str_map_function:nN
24     { \int_to_base:nn {#1} { 12 } } \__silly_numeral_aux:n
25 }
26 \cs_new:Npn \__silly_numeral_aux:n #1
27 { \__silly_step_aux:eN { "\char_uppercase:N #1+1 } \c_true_bool }

```

(End definition for \sillynumeral and others. These functions are documented on page  .)

5.5 Internals

```

\__silly_step_aux:nN
\__silly_step_aux:eN
\__silly_trim_step:nN
28 \cs_new:Npn \__silly_step_aux:nN #1 #2
29 {
30     \cs_if_exist_use:cF { texorpdfstring } { \use_i:nn }
31     { \__silly_trim_step:nN {#1} #2 }
32     {#1}
33 }
34 \cs_generate_variant:Nn \__silly_step_aux:nN { e }
35 \cs_new:Npn \__silly_trim_step:nN #1 #2
36 {
37     <@@=
38     \exp_args:Nee \@silly@place@picture@nn
39     {
40         \bool_if:NT #2
41         {
42             trim = \int_case:nn {#1}
43             {
44                 { 1 } { 57 ~ 0 ~ 42 ~ 0 }
45                 { 2 } { 59 ~ 0 ~ 29 ~ 0 }
46                 { 3 } { 53 ~ 0 ~ 21 ~ 0 }
47                 { 4 } { 62 ~ 0 ~ 11 ~ 0 }
48                 { 5 } { 50 ~ 0 ~ 8 ~ 0 }
49                 { 6 } { 33 ~ 0 ~ 27 ~ 0 }
50                 { 7 } { 45 ~ 0 ~ 13 ~ 0 }
51                 { 8 } { 56 ~ 0 ~ 27 ~ 0 }
52                 { 9 } { 38 ~ 0 ~ 12 ~ 0 }
53                 { 10 } { 7 ~ 0 ~ 43 ~ 0 }

```



```

54         { 11 } { 33 ~ 0 ~ 33 ~ 0 }
55         { 12 } { 57 ~ 0 ~ 7 ~ 0 }
56     }
57     , clip
58 }
59 }
60 { \int_eval:n {#1} }
61 }

(End definition for \__silly_step_aux:nN and \__silly_trim_step:nN.)

```

\@@silly@place@picture@nn

```

62 \DeclareRobustCommand \@@silly@place@picture@nn [2]
63 {
64     \includegraphics
65     [ #1, page=#2, height=\silly@height ]
66     { \silly@picture }
67 }
68 \AtBeginDocument
69 {
70     \cs_if_exist_use:NF \pdfstringdefDisableCommands { \use_none:n }
71     { \cs_set:Npn \@@silly@place@picture@nn #1 #2 {#2} }
72 }
73 \DeclareHookRule { begindocument }
74 { sillypage } { after } { hyperref }

(End definition for \@@silly@place@picture@nn.)

75 \</package>

```

