

Package `innerscript` v. 1.4a Implementation

Conrad Kosowsky

May 2025

`kosowsky.latex@gmail.com`

For easy, off-the-shelf use, type the following in your document preamble and compile using Lua \LaTeX :

```
\usepackage{innerscript}
```

Overview

The `innerscript` package optionally modifies four aspects of \TeX 's automatic math formatting to improve typesetting: (1) it adds extra space around relation and operation symbols in superscripts and subscripts; (2) it removes extra space around `\left-\right` delimiter pairs; (3) it adds extra space after right delimiters in certain situations; and (4) it forces `\left` and `\right` delimiters to completely cover their contents. Using Lua \LaTeX is required.

This file documents the code for the `innerscript` package. It is not a user guide! If you are looking for instructions on how to load `innerscript` and what it can do for your document, see `innerscript-user-guide.pdf`, which is included with the `innerscript` installation and is available on CTAN. Section 1 begins with the package declaration and option processing. Sections 2 and 3 contain the code to adjust spacing between symbols in `\scriptstyle` and `\scriptscriptstyle` respectively, and Section 4 contains the code for adjusting the spacing around `\mathinner` subformulas. Section 5 handles spacing between `\mathclose` and `\mathord` atoms, and section 6 fills out the resizable delimiter heights. Version history appears at the end of the document.

1 Setup

We begin the implementation by declaring the package and defining a general informational macro. The first 60 lines of `innerscript.sty` are comments.

```
61 \NeedsTeXFormat{LaTeX2e}
62 \ProvidesPackage{innerscript}[2025/05/08 v. 1.4a]
63 \def\IS@info#1{\wlog{Package innerscript Info: #1}}
```

Check whether `\Umathordordspacing` is defined. If not, issue an error and `\endinput`. We normally can't print multiple spaces at once, so within a group, we change the space catcode to 12.

```
64 \ifx\Umathordordspacing\@undefined
```

Acknowledgements: Thanks to Sheldon Axler and Clea F. Rees for pointing a bug in a previous version of `innerscript`.

```

65 \bgroup
66 \catcode`\ =12\relax
67 \def\NoLuaTeXError{\GenericError{}}%
68 {\MessageBreak\MessageBreak
69 Package innerscript error:%
70 \MessageBreak\MessageBreak
71 *****\MessageBreak
72 *                *\MessageBreak
73 *   CANNOT LOAD  *\MessageBreak
74 *   INNERSCRIPT  *\MessageBreak
75 *   LuaTeX Needed *\MessageBreak
76 *                *\MessageBreak
77 *****\MessageBreak@gobbletwo}%
78 {See the innerscript package documentation for explanation.}%
79 {I need LuaTeX to make the innerscript package work. It\MessageBreak
80 looks like the current engine is something else, so I\MessageBreak
81 can't load the package file. To use innerscript, please\MessageBreak
82 typeset with LuaLaTeX. To continue without innerscript,\MessageBreak
83 press return.}}%
84 \expandafter\egroup
85 \NoLuaTeXError
86 \AtEndOfPackage{\IS@info{Failed to load \on@line!}}
87 \expandafter\endinput % we \endinput with a balanced conditional
88 \fi

```

The conditionals will encode package option information.

```

89 \newif\ifIS@script
90 \newif\ifIS@scriptscript
91 \newif\ifIS@legacyscript
92 \newif\ifIS@legacyscriptscript
93 \newif\ifIS@inner
94 \newif\ifIS@close
95 \newif\ifIS@cover

```

And set default options.

```

96 \IS@scripttrue
97 \IS@scriptscripttrue
98 \IS@legacyscriptfalse
99 \IS@legacyscriptscriptfalse
100 \IS@innertrue
101 \IS@closetrue
102 \IS@covertrue

```

Now define and process the package options.

```

103 \DeclareOption{script}           {\IS@scripttrue}
104 \IS@legacyscriptfalse}
105 \DeclareOption{scriptscript}    {\IS@scriptscripttrue}
106 \IS@legacyscriptscriptfalse}
107 \DeclareOption{legacy-script}   {\IS@scripttrue}

```

```

108 \IS@legacyscripttrue}
109 \DeclareOption{legacy-scriptscript}{\IS@scriptscripttrue
110 \IS@legacyscriptscripttrue}
111 \DeclareOption{inner}           {\IS@innertrue}
112 \DeclareOption{close}          {\IS@closetrue}
113 \DeclareOption{cover}          {\IS@covertrue}
114 \DeclareOption{no-script}      {\IS@scriptfalse}
115 \IS@legacyscriptfalse}
116 \DeclareOption{no-scriptscript} {\IS@scriptscriptfalse}
117 \IS@legacyscriptscriptfalse}
118 \DeclareOption{no-inner}       {\IS@innerfalse}
119 \DeclareOption{no-close}       {\IS@closefalse}
120 \DeclareOption{no-cover}       {\IS@coverfalse}
121 \ProcessOptions*

```

If the user requested a legacy option for super or subscripts, issue a warning. We store the warning in `\@tempa`, and if both `\ifIS@legacyscript` and `\ifIS@legacyscriptscript` and false, we turn `\@tempa` into `\relax`. The `\@gobbletwo` eats `\on@line` and a period from `\PackageWarning`.

```

122 \def\@tempa{%
123 \AtEndOfPackage{\PackageWarning{innerscript}
124 {I'm ignoring your request for\MessageBreak
125 legacy spacing\on@line\MessageBreak
126 and using the default space\MessageBreak
127 adjustments instead. (The\MessageBreak
128 package options for legacy\MessageBreak
129 spacing are deprecated.)\@gobbletwo}}
130 \ifIS@legacyscript
131 \else
132 \ifIS@legacyscriptscript
133 \else
134 \let\@tempa\relax
135 \fi
136 \fi
137 \@tempa

```

2 Superscripts and Subscripts

First we set the spacing for `\scriptstyle` and `\scriptscriptstyle` atoms. Each space adjustment has the form

$$\Umath\langle classes \rangle \text{spacing}\langle style \rangle \mu \text{expr}\langle skip \rangle * \langle factor \rangle \relax.$$

The $\langle classes \rangle$ are a pair of choices from `ord`, `op`, `bin`, `rel`, `open`, `close`, `punct`, and `inner`, and the $\langle style \rangle$ is either `\scriptstyle`, `\scriptscriptstyle`, or the cramped version of each style. The $\langle skip \rangle$ is the amount of muglue that typically appears between the $\langle classes \rangle$ while in `\textstyle` or `\displaystyle`. The package takes that muglue and scales it by 0.6

Table 1: Space Inserted between Atoms

Consecutive Atom Types	Default space added
<code>\mathord\mathop</code>	<code>\thinmuskip</code>
<code>\mathord\mathbin</code>	<code>\medmuskip</code>
<code>\mathord\mathrel</code>	<code>\thickmuskip</code>
<code>\mathord\mathinner</code>	<code>\thinmuskip</code>
<code>\mathop\mathord</code>	<code>\thinmuskip</code>
<code>\mathop\mathop</code>	<code>\thinmuskip</code>
<code>\mathop\mathrel</code>	<code>\thickmuskip</code>
<code>\mathop\mathinner</code>	<code>\thickmuskip</code>
<code>\mathbin\mathord</code>	<code>\medmuskip</code>
<code>\mathbin\mathop</code>	<code>\medmuskip</code>
<code>\mathbin\mathopen</code>	<code>\medmuskip</code>
<code>\mathbin\mathinner</code>	<code>\medmuskip</code>
<code>\mathrel\mathord</code>	<code>\thickmuskip</code>
<code>\mathrel\mathop</code>	<code>\thickmuskip</code>
<code>\mathrel\mathopen</code>	<code>\thickmuskip</code>
<code>\mathrel\mathinner</code>	<code>\thickmuskip</code>
<code>\mathclose\mathop</code>	<code>\thinmuskip</code>
<code>\mathclose\mathbin</code>	<code>\medmuskip</code>
<code>\mathclose\mathrel</code>	<code>\thickmuskip</code>
<code>\mathclose\mathinner</code>	<code>\thinmuskip</code>
<code>\mathpunct\mathord</code>	<code>\thinmuskip</code>
<code>\mathpunct\mathop</code>	<code>\thinmuskip</code>
<code>\mathpunct\mathrel</code>	<code>\thinmuskip</code>
<code>\mathpunct\mathopen</code>	<code>\thinmuskip</code>
<code>\mathpunct\mathclose</code>	<code>\thinmuskip</code>
<code>\mathpunct\mathpunct</code>	<code>\thinmuskip</code>
<code>\mathpunct\mathinner</code>	<code>\thinmuskip</code>
<code>\mathinner\mathord</code>	<code>\thinmuskip</code>
<code>\mathinner\mathop</code>	<code>\thinmuskip</code>
<code>\mathinner\mathbin</code>	<code>\medmuskip</code>
<code>\mathinner\mathrel</code>	<code>\thickmuskip</code>
<code>\mathinner\mathopen</code>	<code>\thinmuskip</code>
<code>\mathinner\mathpunct</code>	<code>\thinmuskip</code>
<code>\mathinner\mathinner</code>	<code>\thinmuskip</code>

for superscripts and subscripts and by 0.4 for second-order superscripts and subscripts. Table 1 shows the usual amounts of space between consecutive atom types. To actually code it, we define `\@tempa` to expand to various space-adjustment commands depending on package options.

```

138 \ifIS@script           % change \scriptstyle
139 \IS@info{Adjusting space for \string\scriptstyle.}
140 \IS@info{Adjusting space for \string\crampedscriptstyle.}
141 \ifIS@scriptscript % change \scriptscriptstyle
142 \IS@info{Adjusting space for \string\scriptscriptstyle.}
143 \IS@info{Adjusting space for \string\crampedscriptscriptstyle.}

```

In this branch, `\@tempa` expands to space-adjustment commands for all four styles of superscripts and subscripts. The `#1` argument is a `\Umath` spacing primitive, and `#2` is a mu glue register.

```

144 \def\@tempa#1#2{%
145   #1\scriptstyle           \muexpr #2 * 6 / 10\relax
146   #1\crampedscriptstyle   \muexpr #2 * 6 / 10\relax
147   #1\scriptscriptstyle    \muexpr #2 * 4 / 10\relax
148   #1\crampedscriptscriptstyle\muexpr #2 * 4 / 10\relax}
149 \else                       % do not change \scriptscriptstyle
150 \IS@info{No space changes for \string\scriptscriptstyle.}
151 \IS@info{No space changes for \string\crampedscriptscriptstyle.}

```

Now change just first-order superscripts and subscripts.

```

152 \def\@tempa#1#2{%
153   #1\scriptstyle           \muexpr #2 * 6 / 10\relax
154   #1\crampedscriptstyle   \muexpr #2 * 6 / 10\relax}
155 \fi
156 \else                       % do not change \scriptstyle
157 \IS@info{No space changes for \string\scriptstyle.}
158 \IS@info{No space changes for \string\crampedscriptstyle.}
159 \ifIS@scriptscript % change \scriptscriptstyle
160 \IS@info{Adjusting space for \string\scriptscriptstyle.}
161 \IS@info{Adjusting space for \string\crampedscriptscriptstyle.}

```

In this case, change just second-order superscripts and subscripts.

```

162 \def\@tempa#1#2{%
163   #1\scriptscriptstyle    \muexpr #2 * 4 / 10\relax
164   #1\crampedscriptscriptstyle\muexpr #2 * 4 / 10\relax}
165 \else                       % do not change \scriptscriptstyle
166 \IS@info{No space changes for \string\scriptscriptstyle.}
167 \IS@info{No space changes for \string\crampedscriptscriptstyle.}

```

And here change nothing.

```

168 \let\@tempa\@gobbletwo
169 \fi
170 \fi

```

Now actually make the space changes as applicable.

```

171 \@tempa{\Umathordopspacing}   {\thinmuskip}
172 \@tempa{\Umathordbinspacing} {\medmuskip}
173 \@tempa{\Umathordrelspacing}  {\thickmuskip}
174 \@tempa{\Umathordinnerspacing} {\thinmuskip}
175 \@tempa{\Umathopordspacing}   {\thinmuskip}

```

```

176 \@tempa{\Umathopopspacing}      {\thinmuskip}
177 \@tempa{\Umathoprelspacing}     {\thickmuskip}
178 \@tempa{\Umathopinnerspacing}  {\thickmuskip}
179 \@tempa{\Umathbinordspacing}    {\medmuskip}
180 \@tempa{\Umathbinopspacing}     {\medmuskip}
181 \@tempa{\Umathbinopenspacing}   {\medmuskip}
182 \@tempa{\Umathbininnerspacing}  {\medmuskip}
183 \@tempa{\Umathrelordspacing}    {\thickmuskip}
184 \@tempa{\Umathreloppspacing}    {\thickmuskip}
185 \@tempa{\Umathreloppspacing}    {\thickmuskip}
186 \@tempa{\Umathrelinnerspacing}  {\thickmuskip}
187 \@tempa{\Umathcloseopspacing}   {\thinmuskip}
188 \@tempa{\Umathclosebinspacing}  {\medmuskip}
189 \@tempa{\Umathcloserelspacing}  {\thickmuskip}
190 \@tempa{\Umathcloseinnerspacing}{\thinmuskip}
191 \@tempa{\Umathpunctordspacing}  {\thinmuskip}
192 \@tempa{\Umathpunctopspacing}  {\thinmuskip}
193 \@tempa{\Umathpunctrelspacing}  {\thinmuskip}
194 \@tempa{\Umathpunctopspacing}   {\thinmuskip}
195 \@tempa{\Umathpunctclosespacing}{\thinmuskip}
196 \@tempa{\Umathpunctpunctspacing}{\thinmuskip}
197 \@tempa{\Umathpunctinnerspacing}{\thinmuskip}
198 \@tempa{\Umathinnerordspacing}  {\thinmuskip}
199 \@tempa{\Umathinneropspacing}   {\thinmuskip}
200 \@tempa{\Umathinnerbinspacing}  {\medmuskip}
201 \@tempa{\Umathinnerrelspacing}  {\thickmuskip}
202 \@tempa{\Umathinneropspacing}   {\thinmuskip}
203 \@tempa{\Umathinnerpunctspacing}{\thinmuskip}
204 \@tempa{\Umathinnerinnerspacing}{\thinmuskip}

```

3 Mathclose Spacing

In `\textstyle` and `\displaystyle`, we add half a `\thinmuskip` between `\mathclose` and `\mathord` atoms. As with the `script` and `scriptscript` options, we scale the spacing by 0.6 and 0.4 for `\scriptstyle` and `\scriptscriptstyle` respectively. Even though `close` comes after `inner` in the options list in the user guide, it is important to put it before `inner` in the implementation. We should set all interatom spacing before changing the spacing around `\mathinner` subformulas because the `inner` spacing changes reference values of other `\Umath` spacing primitives.

```

205 \ifIS@close
206   \IS@info{Adding space after closing grouping symbols.}
207   \Umathcloseordspacing\displaystyle      \muexpr\thinmuskip / 2\relax
208   \Umathcloseordspacing\textstyle        \muexpr\thinmuskip / 2\relax
209   \Umathcloseordspacing\crampeddisplaystyle \muexpr\thinmuskip / 2\relax
210   \Umathcloseordspacing\crampedtextstyle  \muexpr\thinmuskip / 2\relax
211 \ifIS@script

```

```

212   \Umathcloseordspacing\scriptstyle      \muexpr\thinmuskip * 3 / 10\relax
213   \Umathcloseordspacing\crampedscriptstyle\muexpr\thinmuskip * 3 / 10\relax
214   \fi
215   \ifIS@scriptscript
216   \Umathcloseordspacing\scriptscriptstyle \muexpr\thinmuskip / 5\relax
217   \Umathcloseordspacing\crampedscriptscriptstyle
218   \muexpr\thinmuskip / 5\relax
219   \fi
220 \else
221   \IS@info{No changes to space after delimiters.}
222 \fi

```

4 Mathinner Subformulas

Now set the spacing for `\mathinner` subformulas. Prior to version 1.4, `innerscript` spaced `\mathinner` subformulas under the `inner` option like `\mathord` atoms, but as of version 1.4, `innerscript` treats these subformulas like `\mathopen` on the left and `\mathclose` on the right. Every space adjustment in this section has the form

$$\Umath\langle classes \rangle spacing\langle style \rangle = \the\Umath\langle other classes \rangle spacing\langle style \rangle \relax.$$

In most cases, the $\langle classes \rangle$ are a pair of `inner` and one of `ord`, `op`, `bin`, `rel`, `open`, `close`, or `punct`, and the $\langle other classes \rangle$ are the same except with `inner` replaced with `open` or `close` depending on whether it appears on the left or the right. We can also have $\langle classes \rangle$ be `innerinner`, in which case $\langle other classes \rangle$ is `closeopen`. The $\langle style \rangle$ is one of the eight math style commands: `\displaystyle`, `\textstyle`, `\scriptstyle`, `\scriptscriptstyle`, and the cramped versions. This section comes after `script` and `close` in the implementation because the spacing here depends on the spacing of other math classes.

```

223 \ifIS@inner
224   \IS@info{Adjusting space around \string\mathinner\space subformulas.}

```

First, we loop through the math classes and let `\@i` be a math class.

```

225   \@tfor\@i:=\displaystyle\crampeddisplaystyle
226   \textstyle\crampedtextstyle
227   \scriptstyle\crampedscriptstyle
228   \scriptscriptstyle\crampedscriptscriptstyle
229   \do{%

```

For each style, first set `\Umathinnerinnerspacing`, and then loop through the math classes. We simultaneously set the spacing for atom pairs where `inner` appears either first or second. Using `\expanded` solves a problem where the `\Umath\langle atom \rangle inner` primitives don't behave properly if they appear on both sides of an assignment. I am unclear if this is a bug in LuaTeX?

```

230     \expanded{\Umathinnerinnerspacing\@i=\the\expandafter
231       \Umathcloseopenspacing\@i\relax}
232   \@for\@j:=ord,op,bin,rel,open,close,punct\do{%
233     \expanded{%

```

```

234     \csname Umath\@j innerspacing\endcsname\@i=\the
235     \csname Umath\@j openspacing\expandafter\endcsname\@i\relax
236     \csname Umathinner\@j spacing\endcsname\@i=\the
237     \csname Umathclose\@j spacing\expandafter\endcsname\@i\relax}}
238 \else
239   \IS@info{No changes to space for \string\mathinner\space subformulas.}
240 \fi

```

5 Delimiter Heights

Finally, filling out the delimiter heights is easy. We set `\delimiterfactor` to 1000.

```

241 \ifIS@cover
242   \IS@info{Setting delimiters to full height.}
243   \delimiterfactor\@m
244 \else
245   \IS@info{No changes to delimiter heights.}
246 \fi
247 \wlog{}

```

And that's a wrap!

Version History

- 1.0** February 2021
—initial release
- 1.1** February 2021
—bug fix for `\mathinner` spacing
—added `\IS@skip` muglue register
- 1.2** November 2023
—redesigned spacing for options `script`
and `scriptscript`
—no more `\IS@skip`—options `script`
and `scriptscript` now use `\the`
—added option `close`
—added option `cover`
—separated implementation and user
guide
—added `\scalemu`
- 1.3** August 2024
—bug fix involving `\cramedisplaystyle`
typo
- 1.4** April 2025
—refactored package code
—`legacy-` package options for
superscripts and subscripts now same as
default options
—use `\muexpr` instead of `\scalemu`
—`inner` option changed to look more like
`\mathopen` and `\mathclose` atoms
- 1.4a** May 2025
—bug fix involving `inner` adjustments
not registering properly

Index

- `\l` 66
- C**
- `\crampeddisplaystyle` 209, 225
- `\crampedscriptscriptstyle`
.. 143, 148, 151, 161, 164, 167, 217, 228
- `\crampedscriptstyle`
..... 140, 146, 154, 158, 213, 227
- `\crampedtextstyle` 210, 226
- D**
- `\delimiterfactor` 243
- `\displaystyle` 207, 225
- I**
- `\ifIS@close` 94, 205
- `\ifIS@cover` 95, 241
- `\ifIS@inner` 93, 223
- `\ifIS@legacyscript` 91, 130
- `\ifIS@legacyscriptscript` 92, 132
- `\ifIS@script` 89, 138, 211
- `\ifIS@scriptscript` 90, 141, 159, 215
- `\IS@closefalse` 119
- `\IS@closetrue` 101, 112
- `\IS@coverfalse` 120
- `\IS@covertrue` 102, 113
- `\IS@innerfalse` 118
- `\IS@innetrue` 100, 111
- `\IS@legacyscriptfalse` 98, 104, 115
- `\IS@legacyscriptscriptfalse` 99, 106, 117
- `\IS@legacyscriptscripttrue` 110
- `\IS@legacyscripttrue` 108
- `\IS@scriptfalse` 114
- `\IS@scriptscriptfalse` 116
- `\IS@scriptscripttrue` 97, 105, 109
- `\IS@scripttrue` 96, 103, 107
- M**
- `\mathinner` 224, 239
- `\medmuskip` 172, 179–182, 188, 200
- N**
- `\NoLuaTeXError` 67, 85
- P**
- `\PackageWarning` 123
- S**
- `\scriptscriptstyle` 141, 142, 147,
149, 150, 159, 160, 163, 165, 166, 216, 228
- `\scriptstyle`
.. 138, 139, 145, 153, 156, 157, 212, 227
- T**
- `\textstyle` 208, 226
- `\thickmuskip` 173, 177, 178, 183–186, 189, 201
- `\thinmuskip` 171, 174–176, 187, 190–199,
202–204, 207–210, 212, 213, 216, 218
- U**
- `\Umathbininnerspacing` 182
- `\Umathbinopenspacing` 181
- `\Umathbinopspacing` 180
- `\Umathbinordspacing` 179
- `\Umathclosebinspacing` 188
- `\Umathcloseinnerspacing` 190
- `\Umathcloseopenspacing` 231
- `\Umathcloseopspacing` 187
- `\Umathcloseordspacing`
..... 207–210, 212, 213, 216, 217
- `\Umathcloserelspacing` 189
- `\Umathinnerbinspacing` 200
- `\Umathinnerinnerspacing` 204, 230
- `\Umathinneropenspacing` 202
- `\Umathinneropspacing` 199
- `\Umathinnerordspacing` 198
- `\Umathinnerpunctspacing` 203
- `\Umathinnerrelspacing` 201
- `\Umathopinnerspacing` 178
- `\Umathopopspacing` 176
- `\Umathopordspacing` 175
- `\Umathoprelspacing` 177
- `\Umathordbinspacing` 172
- `\Umathordinnerspacing` 174
- `\Umathordopspacing` 171
- `\Umathordordspacing` 64
- `\Umathordrelspacing` 173
- `\Umathpunctclosespacing` 195
- `\Umathpunctinnerspacing` 197
- `\Umathpunctopenspacing` 194
- `\Umathpunctopspacing` 192
- `\Umathpunctordspacing` 191
- `\Umathpunctpunctspacing` 196

<code>\Umathpunctrelspacing</code>	193	<code>\Umathreloppspacing</code>	184
<code>\Umathrelinnerspacing</code>	186	<code>\Umathrelordspacing</code>	183
<code>\Umathreloppspacing</code>	185		