

Non-Floating Margin Notes with `marginnote` Package*

Markus Kohm[†]

2009/05/06

Abstract

In L^AT_EX the command `\marginpar[⟨left⟩]{⟨right⟩}` might be used to create a note in the margin. But there is a problem with this command: it creates a special kind of float. For this it cannot be used e.g. at floats or footnotes. Package *marginnote* supports another command `\marginnote` to create notes in the margin. This does not use a kind of float and for this does not have the disadvantage of `\marginpar`. But there might be other problems ...

1 How to Use `marginnote` Package

First of all you have to load. You may use:

```
\usepackage{marginnote}
```

to do so.

`\marginnote` The command `\marginnote[⟨left⟩]{⟨right⟩}[⟨voffset⟩]` may be used to set a margin note using `marginnote`. The first optional argument and the mandatory argument are same using `\marginpar` from the L^AT_EX kernel. Even `\reversemarginpar` will be considered. The note `⟨left⟩` or `⟨right⟩` will be put at the current vertical position. Second optional argument `⟨voffset⟩` may be used to adjust the vertical position of the margin note. Use a negative dimension to move it up or a positive dimension to move it down.

`\marginnoteleftadjust`
`\marginnoterightadjust` At some environments, e.g. `framed` from the `framed` package the horizontal placement of the margin notes are not correct. In this case you may redefine `\marginnoteleftadjust` and `\marginnoterightadjust` to fix this. Note that these are macros not lengths! So you have to use `\renewcommand`, `\def` or `\let` to change them. You may e.g. use

```
\begingroup
```

*This file has version number v1.1d, last revised 2009/05/06.

[†]Email: komascript@gmx.info

```

\makeatletter
\g@addto@macro\framed{%
  \let\marginnoteleftadjust\FrameSep
  \let\marginnoterightadjust\FrameSep
}
\endgroup

```

at your preamble after loading package `framed` to fix the problem using `framed` environment.

NOTE: `\marginnoteleftadjust` and `\marginnoterightadjust` will be used only, if the correct horizontal position cannot be determined using PDF_TE_X features (`\pdfsavepos` and `\pdfsavepos`). So if you are using PDF_LA_TE_X with PDF output or PDF_LA_TE_X with PDF_TE_X-version since 1.40 or X_YL_AT_EX you will not need to use the example code above, but you will need at least two L_AT_EX runs to get the correct horizontal positions of the margin notes.

`\marginnotetextwidth`

Package *marginnote* needs to know the real width of the type area to find the right margin. While some environments (e.g. those of package *framed*) change `\textwidth`, *marginnote* defines its own text width macro. If you change type area after `\begin{document}` you should add

```
\edef\marginnotetextwidth{\the\textwidth}
```

after changing the type area. Maybe you should do this globally using `\xdef` instead of `\edef`. Most users will never need to change `\marginnotetextwidth`.

`\marginnotevadjust`

At some environments the vertical adjustment of the margin note will be wrong, e.g. one base line to low. In this case you may use the additional optional argument of `\marginnote` at every usage of `\marginnote` or redefine `\marginnotevadjust` at the begin of the environment. The default definition is `0pt`.

`\raggedleftmarginnote`

`\raggedrightmarginnote`

These macros define how the margin note will be aligned. The defaults are:

- align margin notes at the left margin right to the margin,
- align margin notes at the right margin left to the margin.

You may change this using `\renewcommand`, e.g. use

```

\renewcommand*{\raggedleftmarginnote}{}
\renewcommand*{\raggedrightmarginnote}{\centering}

```

to get justified text at the left and centered text at the right margin.

`\marginfont`

This macro defines the font that will be used to set margin notes. The default is `\normalcolor`. You may use `\renewcommand` to change this, e.g. use

```
\renewcommand*{\marginfont}{\color{red}\sffamily}
```

to get red colored margin notes in sans serif font family. You need to load e.g. package `color` to use `\color`.

2 Known Problems Using `marginnote`

At double side layout (e.g. using class option `twoside`) `\marginnote` needs to know the number of the current page to decide whether the page is odd or even and so whether to use left or right margin. \LaTeX uses an asynchronous output. Because of this counter `page` should not be used to get the number of the current page unless you are at page head or foot. To solve the problem `marginnote` uses a mechanism similar to labels. But this means, that the correct margin won't be known at this \LaTeX run but only at the next. So after adding or deleting a margin note or after each change of page break you need two \LaTeX runs to get all margins right.

The command `\marginnote` uses `\strut` and `\vadjust` to put the margin note at the correct position. But under some circumstances this may fail. You may adjust the vertical position of the margin note using the second optional argument of `\marginnote`. Sometimes even the text outside `\marginnote` will be moved because of using `\marginnote`. There's currently no other solution for this problem than moving the `\marginnote` command.

Note: The margin note will be placed at the current vertical line. This means, if you are using two `\marginnote` commands at the same line, they will be put on the same place. This is not a bug but a feature!

Since release 1.1b `\marginnote` between paragraphs (in vertical mode) will place the note between the paragraphs instead of the end of the previous paragraph. You may use `\leavevmode` or the third optional argument of `\marginnote` to place it different.

No page break may occur inside a margin note created with `\marginnote`.

`\marginnote` somewhat different from `\marginpar` if used immediately after `\item`. This is not a bug, it's a feature!

With math `\marginnote` may work or may not depending on the math environment.

If you are using \XeTeX , \PDFLaTeX since version 1.40 or \PDFLaTeX before version 1.40 with PDF output and the horizontal position of a margin note is wrong, do one more \PDFLaTeX run.

Sometimes lines are stretched vertically using `\marginnote`, e.g. if you're using `\marginnote` at a list *and* upper case umlauts like "Ü" or if `\lineskiplimit>0pt`. In this case `\lineskiplimit=-\maxdimen` should help.

You should not use `\marginnote` at the optional argument of `\item`.

3 Implementation

First test \e-TeX .

```
1 \begingroup
2 \def\@tempb{}%
3 \def\@tempa{%
4   \PackageError{marginnote}{seems you are not running e-TeX\@tempb}{%
5     Since 2004 the LaTeX team recommends to use e-TeX.\MessageBreak
```

```

6      marginnote since version 1.1d uses e-TeX features.\MessageBreak
7      At actual systems 'latex' should already use e-TeX.\MessageBreak
8      At deprecated systems it may be called 'elatex'.\MessageBreak
9      Use either unsupported marginnote up to version 1.1c or\MessageBreak
10     ask you administrator for LaTeX using e-TeX\@tempb.\MessageBreak
11     Not using e-TeX\@tempb\space is a fatal error!\MessageBreak
12     Processing cannot be continued!}%
13 \endgroup
14 \batchmode \errmessage{}\csname @@end\endcsname\end\relax
15 \csname endinput\endcsname
16 }%
17 \expandafter\ifx\csname eTeXversion\endcsname\relax\else
18   \ifnum\eTeXversion <2
19     \def\@tempb{ V 2}%
20   \else
21     \let\@tempa\endgroup
22   \fi
23 \fi
24 \@tempa

```

Next declare and process the options.

```

\if@mn@verbose Use verbose output mode by default. But you may change this using option quiet.
25 \newif\if@mn@verbose\@mn@verbosetrue
26 \DeclareOption{verbose}{\@mn@verbosetrue}
27 \DeclareOption{quiet}{\@mn@verbosetruefalse}

28 \ExecuteOptions{verbose}
29 \ProcessOptions\relax

```

```

\newmarginnote We need a macro to define a new note at the aux file. This will be done using the
mechanism of LATEX that is used for \newlabel. But we use another prefix. This
will result in the usual “Labels(s) may have changed. Rerun to get cross-references
right.” if a margin note is new or have moved to another page.
30 \newcommand*{\newmarginnote}{\@newl@bel{mn}}

```

```

\if@mn@pdfmode We need to know, whether or not PDFTEX and which version of PDFTEX is used.
With PDFTEX the horizontal output position may be detected using \pdfsavepos
and \pdflastxpos. So the relative position of the margin may be calculated.
Without PDFTEX only manual adjustment is available. While PDF mode or not
may change before start of the document, setting up the switch is delayed.
31 \newif\if@mn@pdfmode\@mn@pdfmodefalse
32 \AtBeginDocument{%
33   \begingroup\expandafter\expandafter\expandafter\endgroup
34   \expandafter\ifx\csname pdflastxpos\endcsname\relax\else % bg or 1
35     \begingroup\expandafter\expandafter\expandafter\endgroup
36     \expandafter\ifx\csname pdftexversion\endcsname\relax % bg 2
37       \begingroup\expandafter\expandafter\expandafter\endgroup
38       \expandafter\ifx\csname pdfoutput\endcsname\relax % bg 3
39         \begingroup\expandafter\expandafter\expandafter\endgroup

```

```

40      \expandafter\ifx\csname XeTeXrevision\endcsname\relax\else % bg 4
41      \mn@pdfmodetrue
42      \fi % ed 4
43    \else % or 3
44      \ifcase\pdfoutput\else\mn@pdfmodetrue\fi % bg ed 4
45      \fi % ed 3
46    \else % or 2
47      \ifnum \pdfTeXversion<140 % bg 3
48        \begingroup\expandafter\expandafter\expandafter\endgroup
49        \expandafter\ifx\csname pdfoutput\endcsname\relax % bg 4
50        \else % or 4
51          \ifcase\pdfoutput\else\mn@pdfmodetrue\fi % bg ed 5
52          \fi % ed 4
53        \else % or 3
54          \mn@pdfmodetrue
55          \fi % ed 3
56        \fi % ed 2
57      \fi % ed 1
58    \ifmn@verbose
59      \ifmn@pdfmode
60        \PackageInfo{marginnote}{%
61          \string\pdfoutput\space not 0 and \string\pdflastxpos\space
62          available.\MessageBreak
63          Extended position detection mode activated\@gobble
64        }%
65      \else
66        \PackageInfo{marginnote}{%
67          either \string\pdflastxpos\space or \string\pdfoutput\space not
68          available\MessageBreak
69          or \string\pdfoutput\space set to 0.\MessageBreak
70          Extended position detection mode deactivated\@gobble
71        }%
72      \fi
73    \fi
74 }

```

`\marginnotetextwidth` Some environments change `\textwidth`. But at PDF mode we need to know the real text width to find the right margin. So we use our own text width macro. Sometimes it may be usefull if the user can set it up. Because of this it is a user command.

```

75 \newcommand*{\marginnotetextwidth}{}
76 \let\marginnotetextwidth\textwidth
77 \AtBeginDocument{\ifmn@pdfmode\edef\marginnotetextwidth{\the\textwidth}\fi}

```

`\mn@margintest` Macro `\mn@margintest` does the complete test, which margin to use. The result may be found at `\if@tempwa`. To avoid changes on the last page if there is a new note on the first page, try to count the notes by page. We know that this can not be successfull, but never the less it may be a good try. `\mn@thispage` saves the page number of the last usage of `\mn@margintest`. `\mn@atthispage`

saves the number of margin note at this page. But we need to know the absolut page number to do this. So we increase the absolut page number `mn@abspage` at every `\@outputpage`. `\@mn@currpage` is the page from the page label if found. `\@mn@currxpos` is somehow special. Using PDF_TE_X the real x position may be written with the page label and used to calculate the correct horizontal offset. In this case `\marginnoteleftadjust` and `\marginnoterightadjust` will not be used!

```

78 \newcommand*{\@mn@thispage}{%
79 \newcommand*{\@mn@currpage}{%
80 \newcommand*{\@mn@currxpos}{%
81 \newcounter{mn@abspage}
82 \AtBeginDocument{\setcounter{mn@abspage}{1}%
83 \g@addto@macro\@outputpage{\stepcounter{mn@abspage}}}%
84 \newcommand*{\@mn@margintest}{%

```

Number of the next margin note at this page.

```

85 \expandafter\ifx\csname @mn@thispage\endcsname\@empty
86 \gdef\@mn@atthispage{1}%
87 \else\expandafter\ifnum \@mn@thispage=\value{mn@abspage}%
88 \begin{group}
89 \tempcnta\@mn@atthispage\advance\tempcnta by \@ne
90 \xdef\@mn@atthispage{\the\tempcnta}%
91 \end{group}
92 \else
93 \gdef\@mn@atthispage{1}%
94 \fi
95 \fi
96 \xdef\@mn@thispage{\themn@abspage}%

```

Use the number of the page and the number of the margin note at this page to save the real number of this page at the aux file. At PDF mode save the current x position too.

```

97 \let\@mn@currpage\relax
98 \let\@mn@currxpos\relax
99 \if@mn@pdfmode
100 \pdfsavepos
101 \protected@write\@auxout{\let\themn@abspage\relax}{%
102 \string\newmarginnote{note.\@mn@thispage.\@mn@atthispage}{%
103 {\themn@abspage}{\noexpand\number\pdflastxpos sp}}}%
104 }%
105 \else
106 \protected@write\@auxout{\let\themn@abspage\relax}{%
107 \string\newmarginnote{note.\@mn@thispage.\@mn@atthispage}{%
108 {\themn@abspage}{}}}%
109 }%
110 \fi

```

If the margin note label was not defined, it seems to be new. In this case the absolut page number will be used for the test instead of the saved real page number.

```

111 \expandafter\ifx\csname mn@note.\@mn@thispage.\@mn@atthispage\endcsname\relax
    If we are not in two side mode, we are on a odd page.
112 \if@twoside
113 \if@mn@verbose
114 \PackageInfo{marginnote}{Suggest that margin
115 note \@mn@thispage.\@mn@atthispage\space will be on\MessageBreak
116 absolute page \themn@abspage.\MessageBreak
117 This may be wrong}%
118 \fi
119 \ifodd\value{mn@abspage}\@tempwattrue\else\@tempwafalse\fi
120 \else
121 \if@mn@verbose
122 \PackageInfo{marginnote}{right page because not two side mode}%
123 \fi
124 \@tempwattrue
125 \fi
126 \else
127 \edef\@mn@currpage{\csname
128 mn@note.\@mn@thispage.\@mn@atthispage\endcsname}%
129 \edef\@mn@currxpos{\expandafter\@secondoftwo\@mn@currpage}%
    Ulrike Fischer suggested a simple change to take care of \hoffset, e.g., using
    package crop. We use this occasion to take care of \pdfhorigin, too.
130 \edef\@mn@currxpos{\the\dimexpr \@mn@currxpos -\hoffset\relax}%
131 \begingroup\expandafter\expandafter\expandafter\endgroup
132 \expandafter\ifx\csname pdfhorigin\endcsname\relax\else
133 \begingroup\expandafter\expandafter\expandafter\endgroup
134 \expandafter\ifx\csname pdfoutput\endcsname\relax\else
135 \ifnum \pdfoutput=1 %
136 \edef\@mn@currxpos{\the\dimexpr \@mn@currxpos -\pdfhorigin
137 +1in\relax}%
138 \fi
139 \fi
140 \fi
141 \edef\@mn@currpage{\expandafter\@firstoftwo\@mn@currpage}%
142 \if@mn@verbose
143 \PackageInfo{marginnote}{Margin note \@mn@thispage.\@mn@atthispage\space
144 is on absolute page \@mn@currpage\MessageBreak}%
145 \fi
146 \if@twoside
147 \ifodd\@mn@currpage\relax
148 \@tempwattrue
149 \else
150 \@tempwafalse
151 \fi
152 \else
153 \if@mn@verbose
154 \PackageInfo{marginnote}{right page because not two side mode}%
155 \fi
156 \@tempwattrue

```

```

157   \fi
158   \fi
159 }

```

```

\marginnote Command \marginnote is the main macro of the package. The others are helpers
\mn@marginnote to manage the optional arguments.
\mn@@marginnote 160 \newcommand*\marginnote{%
\mn@@@marginnote 161   \@dblarg\mn@marginnote
162 }
163 \newcommand{\mn@marginnote}[2] [] {%
164   \ifhmode
165     \bsphack
166     \begingroup
167     \ifdim\@savsk>\z@ \else
168       \def\:\@xifnch\expandafter\def\: { \futurelet\@let@token\@ifnch}%
169       \fi
170     \else
171       \begingroup
172       \fi
173     \ifnextchar [{\mn@@marginnote[#1]{#2}}{\mn@@marginnote[#1]{#2}[\z@]}%
174 }
175 \newcommand{\mn@@marginnote}{}
176 \def\mn@@marginnote[#1]#2[#3]{%
177   \endgroup

```

In horizontal mode the space hack of the L^AT_EX kernel will be used. In vertical mode this should not be used.

```

178   \ifhmode
179     \mn@@@marginnote[#1]{#2}[\z@]#3}%
180     \esphack
181   \else
182     \mn@@@marginnote[#1]{#2}[\z@]#3}%
183   \fi
184 }
185 \newcommand{\mn@@@marginnote}{}
186 \def\mn@@@marginnote[#1]#2[#3]{%

```

All changes (but change of counters that are global because of using the L^AT_EX commands to change them an `\gdef` and `\xdef`) should be local. In h-mode a `\strut` will be used to fix base line. The margin note will be put to vertical list using `\vadjust`. This also means that we are one line too deep. This will be corrected later using negative kern. In v-mode we use a special kind of vbox to simply set everything. Math mode should behave like v-mode. And if we are just after an item we have to leave v-mode first.

```

187   \begingroup
188     \ifmmode\strut\let\@tempa\mn@vadjust\else
189       \if@inlabel\leavevmode\fi
190     \ifhmode\strut\let\@tempa\mn@vadjust\else\let\@tempa\mn@vlap\fi
191     \fi
192     \@tempa{%

```


Everything will be put upwards using a vbox with zero height and depth and \vss. At this box the margin test will be done. If csreversemargin was used, the logic switches. Then the note will be places to the margin.

```

193      \vbox to\z{%
194      \vss
195      \@mn@margintest
196      \if@reversemargin\if@tempswa
197      \@tempswafalse
198      \else
199      \@tempswatrue
200      \fi\fi
201      \if@tempswa
202      \rlap{%

```

If \@mn@currpos is neither \relax nor empty it is the real current x position of the last PDF^LTeX run and may be used to calculate the real horizontal offset.

```

203      \ifx\@mn@currpos\relax
204      \kern\marginnoterightadjust
205      \if@mn@verbose
206      \PackageInfo{marginnote}{%
207      xpos not known,\MessageBreak
208      using \string\marginnoterightadjust}%
209      \fi
210      \else\ifx\@mn@currpos\empty
211      \kern\marginnoterightadjust
212      \if@mn@verbose
213      \PackageInfo{marginnote}{%
214      xpos not known,\MessageBreak
215      using \string\marginnoterightadjust}%
216      \fi
217      \else
218      \if@mn@verbose
219      \PackageInfo{marginnote}{%
220      xpos seems to be \@mn@currpos,\MessageBreak
221      \string\marginnoterightadjust
222      \space ignored}%
223      \fi
224      \begingroup
225      \setlength{\@tempdima}{\@mn@currpos}%
226      \kern-\@tempdima
227      \if@twoside\ifodd\@mn@currpage\relax
228      \kern\oddsidemargin
229      \else
230      \kern\evensidemargin
231      \fi
232      \else
233      \kern\oddsidemargin
234      \fi
235      \kern 1in
236      \endgroup

```

```

237         \fi
238     \fi
239     \kern\marginnotetextwidth\kern\marginparsep
240     \vbox to\z@{\kern\marginnotevadjust\kern #3
241         \vbox to\z@{%
242             \hsize\marginparwidth

```

Here's the correction of the vertical position. The rest is simple.

```

243         \kern-\parskip
244         \marginfont\raggedrightmarginnote\hspace{\z@}\strut#2\endgraf
245         \vss}%
246     \vss}%
247 }%
248 \else
249     \llap{%
250         \vbox to\z@{\kern #3
251         \vbox to\z@{%
252             \hsize\marginparwidth

```

Same like above for left margins.

```

253         \kern-\parskip
254         \marginfont\raggedleftmarginnote\hspace{\z@}\strut#1\endgraf
255         \vss}%
256     \vss}%
257 \ifx\@mn@currxpos\relax
258     \kern\marginnoteleftadjust
259     \if@mn@verbose
260         \PackageInfo{marginnote}{%
261             xpos not known,\MessageBreak
262             using \string\marginnoteleftadjust}%
263     \fi
264 \else\ifx\@mn@currxpos\@empty
265     \kern\marginnoteleftadjust
266     \if@mn@verbose
267         \PackageInfo{marginnote}{%
268             xpos not known,\MessageBreak
269             using \string\marginnoteleftadjust}%
270     \fi
271 \else
272     \if@mn@verbose
273         \PackageInfo{marginnote}{%
274             xpos seems to be \@mn@currxpos,\MessageBreak
275             \string\marginnoteleftadjust
276             \space ignored}%
277     \fi
278 \begingroup
279     \kern\@mn@currxpos
280     \if@twoside\ifodd\@mn@currpage\relax
281         \kern-\oddsidemargin
282     \else
283         \kern-\evensidemargin

```

```

284             \fi
285             \else
286             \kern-\oddsidemargin
287             \fi
288             \kern-1in
289             \endgroup
290         \fi
291     \fi
292     \kern\marginparsep
293 }%
294 \fi
295 }%
296 }%
297 \endgroup
298 }

```

`\marginnoterightadjust` These may be used to define an automatic horizontal adjust. The default is zero.
`\marginnoteleftadjust` They will be used only if not PDF_TE_X or PDF_TE_X before version 1.40 in DVI mode is used, because in this case the save position features are not available.

```

299 \newcommand*{\marginnoterightadjust}{}
300 \newcommand*{\marginnoteleftadjust}{}
301 \let\marginnoterightadjust\z@
302 \let\marginnoteleftadjust\z@

```

`\marginnotevadjust` This may be used to define an automatic vertical adjust. The default is zero. Values greater than zero will move the margin note down, values less than zero will move the margin note up.

```

303 \newcommand*{\marginnotevadjust}{}
304 \let\marginnotevadjust\z@

```

`\mn@vlap` This macro is used to set a vertical box without size at vertical mode.

```

305 \newcommand{\mn@vlap}[1]{%
306   \setbox\@tempboxa\vbox to \ht\strutbox{#1\vss}%
307   \box\@tempboxa\vskip-\baselineskip
308 }

```

`\mn@zbox` This macro is used to set a horizontal box without height, depth and width.

```

309 \newcommand{\mn@zbox}[1]{%
310   \bgroup
311   \setbox\@tempboxa\vbox{#1}%
312   \ht\@tempboxa\ht\strutbox
313   \dp\@tempboxa\dp\strutbox
314   \wd\@tempboxa\z@
315   \box\@tempboxa
316   \egroup
317 }

```

`\mn@vadjust` This macro is used to set a vertical box at horizontal mode.

```

318 \newcommand{\mn@vadjust}[1]{%

```

```

319 \mn@zbox{\kern-\parskip
320 \leavevmode\vadjust{#1}%
321 \kern\parskip
322 }%
323 }

```

`\marginfont` These are very simple. A class may also define `\marginfont`. Use this if available.

`\raggedleftmarginnote` I don't use `\let` for the definitions of the ragged macros, so the meaning may

`\raggedrightmarginnote` change loading e.g. package `ragged2e`.

```

324 \providecommand*\marginfont{}{}
325 \newcommand*\raggedleftmarginnote{\raggedleft}
326 \newcommand*\raggedrightmarginnote{\raggedright}

```

Change History

v1.0a		<code>\marginnotetextwidth:</code>	new
General: Example to macros		macro	5
<code>\raggedleftmarginnote</code> and	v1.1a	<code>\if@mn@pdfmode:</code> PDF _T _E X since	
<code>\raggedrightmarginnote</code> at		1.40 allows <code>\pdfsavepos</code> in DVI	
documentation fixed [thanks to		mode too	4
Susumu Tanimura].	v1.1b	<code>\@mn@@@marginnote:</code> use <code>\mn@vadjust</code>	
<code>\marginfont:</code> Use <code>\providecommand</code>		instead of <code>\vadjust</code>	8
to define it.	v1.0b	<code>\if@mn@pdfmode:</code> if level fixed	4
12		<code>\mn@vadjust:</code> new (internal)	11
v1.0b		<code>\mn@zbox:</code> new (internal)	11
General: spelling fixes	v1.1		
1		<code>\@mn@@@marginnote:</code> new PDF	
v1.1		mode feature	8
<code>\@mn@@@marginnote:</code> new PDF	v1.1c	<code>\if@mn@pdfmode:</code> X _Y TeX has work-	
mode feature		ing <code>\pdflastxpos</code> but no	
<code>\@mn@currpage:</code> new (internal) ...		<code>\pdftexversion</code>	4
5	v1.1d	<code>mn@abspage:</code> take care of <code>\hoffset</code>	7
<code>\@mn@currxpos:</code> new (internal) ...		take care of <code>\pdfhorigin</code>	7
5			
<code>\@mn@margintest:</code> new PDF mode			
feature			
5			
<code>\if@mn@pdfmode:</code> new switch			
4			