

physicx, 扩展的 physics 宏包

雾月, Longaster*

2021 年 11 月 12 日 v0.1.2

目录

第 1 节 简介	1	4.4 genegralmatrix	11
第 2 节 基本用法与宏包选项	1	4.5 matrix 类通用键值选项	11
第 3 节 括号, quantity	2	4.6 例子	13
3.1 新的括号命令	2	第 5 节 杂项	16
3.2 旧的括号命令	3	5.1 定义新的括号命令	17
第 4 节 矩阵, matrix	5	5.2 定义新的矩阵命令	18
4.1 diagonalmatrix	6	版本历史	20
4.2 commamatrix	8	代码索引	20
4.3 qxmatrix	9		

第 1 节 简介

physics 宏包定义了一些简写的命令, 但是已经 9 年未更新了, physicx 对其进行了一定的扩展, 基本兼容 physics 原有的命令。

要求: expl3 宏包版本至少 Released 2021-08-27, 不准兼容低版本, 尽管并不依赖某些新特性。如果使用 TeXLive, 可以使用自带的控制台进行更新。如果你想兼容低版本, 可以在 LPPL 1.3c 协议下自行修改。

第 2 节 基本用法与宏包选项

太长不看版: 若仍要使用 physics 宏包, 则使用 physics 宏包选项, 不可再加载 physics 宏包, 否则使用 short 宏包选项。

通常情况下, 可用像使用 physics 宏包那样使用 physicx 的命令, 它们基本兼容的, 但是 physicx 宏包并不依赖 physics 宏包, 但目前 physicx 并不提供诸如 \dd 的命令, 你可以使用 physics 宏包选项来加载 physics 宏包。也可以自行加载, 但必须首先加载 physics 再加载 physicx 宏包, 一般情况下使用 physics 宏包选项即可。

宏包选项: compat 将尽量兼容 physics 宏包。physics: 自动启用 compat 选项并加载 physics 宏包。short: 将重定义一些简写的命令。mathtools: 启用 mathtools 宏包定义的一些环境, 如 matrix* 等。unimath: 加载 unicode-math 宏包并设置一些兼容操作。

reqty 控制是否重定义 physics 宏包的括号类命令。noqty 相当于 reqty=false。初始为重定义。

*Email: longaster@163.com

第3节 括号, *quantity*

3.1 新的括号命令

“新的括号命令”指 `physicsx` 宏包定义的括号命令。

<code>\xquantity</code>	<code>\xquantity {⟨<i>quantity</i>类键值选项⟩} {⟨<i>code</i>⟩}</code>
<code>\qxqty</code>	<code>[⟨<i>quantity</i>类键值选项⟩] {⟨<i>code</i>⟩}</code>
<code>\qxqty *</code>	<code>[⟨<i>quantity</i>类键值选项⟩] {⟨<i>code</i>⟩}</code>

这些命令用于替代 `physics` 宏包的括号类命令。

带 `*` 的 `\qxqty` 命令禁止自动调整括号大小。

`\xquantity` 的速度更快些。

例 1:

```
$ \xquantity{x^2} \quad \xquantity{type=v}{\int} \quad \displaystyle
\xquantity{type=v}{\int_a^b} $ \quad
$ \xquantity{size=\bigg}{x^2} \quad \displaystyle
\xquantity{size=\bigg}{\int_a^b} $
\[
\xquantity{x^2} \quad \quad \xquantity{type=v}{\int} \quad \quad
\xquantity{type=v}{\int_a^b} \quad \quad \xquantity{size=\bigg}{x^2}
\xquantity{size=\bigg}{\int_a^b}
\]

$ \qxqty{x^2} \quad \quad \qxqty{type=v}{\int} \quad \quad \displaystyle
\xqty[v]{\int_a^b} $ \quad
$ \qxqty[size=\bigg]{x^2} \quad \quad \displaystyle
\xqty[size=\bigg]{\int_a^b}$
\[
\xqty{x^2} \quad \quad \qxqty{type=v}{\int} \quad \quad \displaystyle
\xqty[v]{\int_a^b} \quad \quad \qxqty[size=\bigg]{x^2} \quad \quad
\xqty[size=\bigg]{\int_a^b}
\]
```

$$\begin{array}{ccc}
 (x^2) & |f| & \left| \int_a^b \right| & (x^2) & \left(\int_a^b \right) \\
 & & & & \\
 & & & (x^2) & |f| & \left| \int_a^b \right| & (x^2) & \left(\int_a^b \right) \\
 & & & & & & & \\
 (x^2) & |f| & \left| \int_a^b \right| & (x^2) & \left(\int_a^b \right) \\
 & & & & & & & \\
 & & & (x^2) & |f| & \left| \int_a^b \right| & (x^2) & \left(\int_a^b \right)
 \end{array}$$

`physicsx` 宏包的一个标准的括号命令由如下 8 个部分构成:

<code>⟨<i>pre</i>⟩</code> <code>⟨<i>left-size</i>⟩</code> <code>⟨<i>left</i>⟩</code> <code>⟨<i>arg</i>⟩</code> <code>⟨<i>code</i>⟩</code> <code>⟨<i>right-size</i>⟩</code> <code>⟨<i>right</i>⟩</code> <code>⟨<i>post</i>⟩</code>

每一部分都有一个对应的键,可以在 `⟨quantity 类键值选项⟩` 中设置。

pre	它们代表了标准括号命令的对应部分。当 left-size、right-size 有一个为空时,则会移除
post	left-size、right-size 这两个部分。
left	left-size 初始为 \left, right-size 初始为 \right。
right	
left-size	
right-size	
code	
size	size 同时设置 left-size 和 right-size。noauto 将 size 设置为空。一般情况下,可以不
noauto	写 size=,而只写它的值。如 \xqty[size=\Big]{x} 与 \xqty[\Big]{x} 效果相同。
args	args 键将 ⟨arg⟩ 部分设置为 [⟨args⟩],而 args* 键将 ⟨arg⟩ 部分设置为其值。
args*	
type	type = ⟨p b B v V a m pm bm Bm vm Vm sm ...⟩ 初始值 = p
	type 键使用预定义的括号类型。
	它们的值正如 amsmath 的矩阵环境的 matrix 前的部分。p 代表 (), b 代表 [], B 代表 { }, v 代表 “ ”, V 代表 “ ”, a 代表 “⟨⟩”。
	带有 m 的则为矩阵类的括号。
	如果使用了 mathtools 宏包或使用了 mathtools 宏包选项,则还定义了 p*、sm*、spm、spm* 等值,它们对应 pmatrix*、smallmatrix*、psmallmatrix、psmallmatrix* 等环境。可以使用 args 键来设置它们的参数。mathtools 宏包必须手动加载。
	它们是通过 left、right、size 键来设置的。
	注意: left、right 的设置必须合法,即若 size 不为空,则 left、right 必须可以跟在 \left、\right 这两个调整大小的命令后面。不能如同:
	left=\begin{matrix} \end{matrix}, right=\end{matrix}, left-size=\left, right-size=\right。
	一般情况下,可以不写 type= 而只写它的值。但是矩阵类的值则不能这样做,原因目前未知。

例 2:

```

\[
\xquantity{left=\begin{pmatrix}, right=\end{pmatrix}, noauto}
{ ABC & dec \\ hij & HJK } \quad
\xquantity{left=\begin{smallmatrix}, right=\end{smallmatrix}, noauto}
{ ABC & dec \\ hij & HJK } \quad
\xquantity {type=m*} { ABC & dec \\ hij & HJK } \quad
\xquantity {type=sm} { ABC & dec \\ hij & HJK }
\]

```

$$\begin{pmatrix} ABC & dec \\ hij & HJK \end{pmatrix} \quad \begin{smallmatrix} ABC & dec \\ hij & HJK \end{smallmatrix} \quad \begin{matrix} ABC & dec \\ hij & HJK \end{matrix} \quad \begin{matrix} ABC & dec \\ hij & HJK \end{matrix}$$

3.2 旧的括号命令

“旧的括号命令”是指 physics 宏包定义的命令。可以使用 reqty 宏包选项设置是否使用新的机制重新定义它们。

在仅使用 physicsx 而不使用 physics 宏包时,本节命令不宜再使用,仅出于兼容性原因而保留。

增加了 \oorder、\Oorder 命令,以输出 $o(\blacksquare)$, $\mathcal{O}(\blacksquare)$, 它们的大小写总是固定的。而 \order 则会依据 physics 宏包是否加载、compat 选项是否给定来使用大小写,若给定,出于兼容性考虑, \order 输出大写的 $\mathcal{O}(\blacksquare)$ 。

Full	Short	Description
<code>\quantity</code>	$\backslash\text{qty}(\backslash\text{typical})_{\square} \rightarrow (\square)$ $\backslash\text{qty}(\backslash\text{tall})_{\square} \rightarrow (\square)$ $\backslash\text{qty}(\backslash\text{grande})_{\square} \rightarrow (\square)$ $\backslash\text{qty}[\backslash\text{typical}]_{\square} \rightarrow [\square]$ $\backslash\text{qty} \backslash\text{typical} _{\square} \rightarrow \square $ $\backslash\text{qty}\{\backslash\text{typical}\}_{\square} \rightarrow \{\square\}$ $\backslash\text{qty}\langle\backslash\text{typical}\rangle_{\square} \rightarrow \langle\square\rangle$ $\backslash\text{qty}=\backslash\text{typical}=\square \rightarrow \ \square\ $ $\backslash\text{qty}\backslash\text{big}\{\}_{\square} \rightarrow \{\}$ $\backslash\text{qty}\backslash\text{Big}\{\}_{\square} \rightarrow \{\}$ $\backslash\text{qty}\backslash\text{bigg}\{\}_{\square} \rightarrow \{\}$ $\backslash\text{qty}\backslash\text{Bigg}\{\}_{\square} \rightarrow \{\}$ $\backslash\text{pqty}\{\} \leftrightarrow \backslash\text{qty}()$ $\backslash\text{bqty}\{\} \leftrightarrow \backslash\text{qty}[]$ $\backslash\text{vqty}\{\} \leftrightarrow \backslash\text{qty} $ $\backslash\text{Bqty}\{\} \leftrightarrow \backslash\text{qty}\{\}$	<p>automatic () braces</p> <p>automatic [] braces</p> <p>automatic braces</p> <p>automatic { } braces</p> <p>automatic < > braces</p> <p>automatic braces</p> <p>manual sizing (works with any of the above bracket types)</p> <p>alternative syntax; robust and more \LaTeX-friendly</p>
<code>\absolutevalue</code>	$\backslash\text{abs}\{a\} \rightarrow a $ $\backslash\text{abs}\backslash\text{Big}\{a\} \rightarrow a $	<p>automatic sizing; equivalent to $\backslash\text{qty} a$</p> <p>inherits manual sizing syntax from $\backslash\text{qty}$</p>
<code>\norm</code>	$\backslash\text{abs}*\backslash\text{grande} \rightarrow \square $ $\backslash\text{norm}\{a\} \rightarrow \ a\ $ $\backslash\text{norm}\backslash\text{Big}\{a\} \rightarrow \ a\ $ $\backslash\text{norm}*\backslash\text{grande} \rightarrow \ \square\ $	<p>star for no resize</p> <p>automatic sizing</p> <p>manual sizing</p> <p>star for no resize</p>
<code>\evaluated</code>	$\backslash\text{eval}\{x\}_0^{\infty} \rightarrow x _0^{\infty}$ $\backslash\text{eval}\backslash\text{Big}\{x\}_0^{\infty} \rightarrow x _0^{\infty}$ $\backslash\text{eval}(x _0^{\infty} \rightarrow (x _0^{\infty}$ $\backslash\text{eval}[x _0^{\infty} \rightarrow [x _0^{\infty}$ $\backslash\text{eval}[\backslash\text{venti} _0^{\infty} \rightarrow [\square]_0^{\infty}$ $\backslash\text{eval}*\backslash\text{venti} _0^{\infty} \rightarrow [\square]_0^{\infty}$	<p>vertical bar for evaluation limits</p> <p>alternate form</p> <p>alternate form</p> <p>automatic sizing</p> <p>star for no resize</p>
<code>\order</code>	$\backslash\text{order}\{x^2\} \rightarrow \mathcal{O}(x^2)$ $\backslash\text{order}\backslash\text{Big}\{x^2\} \rightarrow \mathcal{O}(x^2)$ $\backslash\text{oorder}\{x^2\} \rightarrow o(x^2)$ $\backslash\text{Oorder}\{x^2\} \rightarrow \mathcal{O}(x^2)$ $\backslash\text{order}*\backslash\text{grande} \rightarrow \mathcal{O}(\square)$	<p>order symbol; automatic sizing and space handling</p> <p>manual sizing</p> <p>lowercase o</p> <p>uppercase O</p> <p>star for no resize</p>
<code>\commutator</code>	$\backslash\text{comm}\{A\}\{B\} \rightarrow [A, B]$ $\backslash\text{comm}\backslash\text{Big}\{A\}\{B\} \rightarrow [A, B]$	<p>automatic sizing</p> <p>manual sizing</p>

<code>\comm*{A}{\grande}</code>	$\rightarrow [A, \blacksquare]$	star for no resize	
<code>\anticommutator</code>	<code>\acomm{A}{B}</code>	$\rightarrow \{A, B\}$	same as <code>\poissonbracket</code>
<code>\poissonbracket</code>	<code>\pb{A}{B}</code>	$\rightarrow \{A, B\}$	same as <code>\anticommutator</code>

出于兼容性考量, 仍然保留了 `\matrixquantity` (`\mqty`)、`\smallmatrixquantity` (`\smqty`) 命令的定义, 它们与 `\quantity` 命令相似, 只是将它们的值放入相应的矩阵中。可以使用更高级的矩阵命令, 见第 4 节。

你还可以自定义 `quantity` 类命令, 见第 5 节。

第 4 节 矩阵, *matrix*

`physicx` 宏包极大地扩展了 `physics` 原有的矩阵类命令。提供了丰富的键值接口来设置矩阵。

定义了三个基本命令: `\genegralmatrix`、`\commamatrix`、`\diagonalmatrix`, 及 `\qxmatri`。

例 3: `\diagonalmatrix`

```

 $\diagonalmatrix\{1,2,3,4\}$,$ 
 $\diagonalmatrix[p]\{1,2,3,4\}$,$ 
 $\diagonalmatrix[p]\{0=\{1,2,3,4\}, 2=\{22,33\}, '-2=\{-22,-33\}\}$,$ 
 $\diagonalmatrix[p,empty=\Box]\{0=\{1,2,3,4\}, 2=\{22,33\}, '-2=\{-22,-33\}\}$,$ 

 $\diagonalmatrix[p,empty=\Box,$ 
   $item=\{\{4\}\{4\}=\blacksquare, \{\{2\}\{3\}=\oplus\}\},$ 
 $]\{0=\{1,2,3,4\}, 2=\{22,33\}, '-2=\{-22,-33\}\}$,$ 
 $\diagonalmatrix[transpose,p,empty=\Box]\{0=\{1,2,3,4\}, 2=\{22,33\}, '-2=\{-22,-33\}\}$,$ 
 $\diagonalmatrix[transpose,p,empty=\Box,$ 
   $item=\{\{4\}\{4\}=\blacksquare, \{\{2\}\{3\}=\oplus\}\},$ 
 $]\{0=\{1,2,3,4\}, 2=\{22,33\}, '-2=\{-22,-33\}\}$,$ 

```

$$\begin{matrix} 1 & & & \\ & 2 & & \\ & & 3 & \\ & & & 4 \end{matrix} \cdot \begin{pmatrix} 1 & & & \\ & 2 & & \\ & & 3 & \\ & & & 4 \end{pmatrix} \cdot \begin{pmatrix} 1 & & 22 & \\ & 2 & & 33 \\ & & 3 & -22 \\ & & -33 & 4 \end{pmatrix} \cdot \begin{pmatrix} 1 & \square & 22 & \square \\ \square & 2 & \square & 33 \\ \square & \square & 3 & -22 \\ \square & \square & -33 & 4 \end{pmatrix}$$

$$\begin{pmatrix} 1 & \square & 22 & \square \\ \square & 2 & \oplus & 33 \\ \square & \square & 3 & -22 \\ \square & \square & -33 & \blacksquare \end{pmatrix} \cdot \begin{pmatrix} 1 & \square & \square & \square \\ \square & 2 & \square & \square \\ 22 & \square & 3 & -33 \\ \square & 33 & -22 & 4 \end{pmatrix} \cdot \begin{pmatrix} 1 & \square & \square & \square \\ \square & 2 & \square & \square \\ 22 & \oplus & 3 & -33 \\ \square & 33 & -22 & \blacksquare \end{pmatrix}$$

例 4: \commamatrix

```

 $\backslash\commamatrix{1, 2, 3, 4}\$, \quad$ 
 $\backslash\commamatrix[b]{1, 2, 3, 4}\$, \quad$ 
 $\backslash\commamatrix{1; 2; 3; 4}\$, \quad$ 
 $\backslash\commamatrix[b]{1; 2; 3; 4}\$, \quad$ 
 $\backslash\commamatrix[b]{1, 2; 3, 4}\$, \quad$ 
 $\backslash\commamatrix[b]{1 \& 2 \\\ 3, 4; 5, 6}\$, \quad$ 

```

$$1 \quad 2 \quad 3 \quad 4, \quad \begin{bmatrix} 1 & 2 & 3 & 4 \end{bmatrix}, \quad \begin{matrix} 1 \\ 2 \\ 3' \\ 4 \end{matrix}, \quad \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, \quad \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix},$$

例 5: \generalmatrix

```

 $\backslash\generalmatrix{p}{A}{m}{n}\$,$ 
 $\backslash\generalmatrix{v}{A}{3}{3}\$,$ 
 $\backslash\generalmatrix*{p}{A}{3}{3}\$,$ 

```

$$\begin{pmatrix} A_{11} & A_{12} & A_{13} & \cdots & A_{1n} \\ A_{21} & A_{22} & A_{23} & \cdots & A_{2n} \\ A_{31} & A_{32} & A_{33} & \cdots & A_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ A_{m1} & A_{m2} & A_{m3} & \cdots & A_{mn} \end{pmatrix}, \quad \begin{vmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{vmatrix}, \quad \begin{pmatrix} A_{11} & A_{12} & A_{13} & \cdots \\ A_{21} & A_{22} & A_{23} & \cdots \\ A_{31} & A_{32} & A_{33} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix},$$

4.1 diagonalmatrix

diagonalmatrix, 即对角矩阵。

```

\diagonalmatrix    [matrix类键值选项] {diag键选项}
\diagonalmatrix =  [matrix类键值选项] {diag键选项}
\diagonalmatrix +  [matrix类键值选项] {diag键选项}
\diagonalmatrix = + [matrix类键值选项] {diag键选项}

```

= 开启保存模式, 即不输出计算的矩阵, 而将其全局保存至 \physicxtmp 宏中, 将其扩展一次即可得到矩阵的值 (不包括外部的 \begin{matrix} 等)。

= 将覆盖 saveto 和 saveto* 键的设置, 即总是将矩阵保存到 \physicxtmp 中, 而不管 saveto 和 saveto* 的设置。

+ 启用 enhanced 模式, 在该模式下能使用更多的键值选项, 但速度相较而言慢些。

diag 键选项:

```

auto-update
noauto-update
true
false
0
1
-1
'1
'-1
...

```

auto-update 将自动更新矩阵的行数和列数, 使得给出的元素总是完整地出现在矩阵中。

true 在设置主对角线元素的同时设置矩阵的行数和列数为主对角线元素个数。行数和列数可以被后续的处理更改。

数字键将设置矩阵的对角元素的值。对角元素的位置是当前矩阵的相应元素的位置。

不带“'”的设置主对角线元素, 即“\”, 带“'”设置副对角线元素, 即“/”。正数为上方, 负数为下方。

注意副对角线元素的设置依赖矩阵的大小, 即矩阵的行数和列数, 在设置副对角线元素时行数和列数必须被直接或间接地设置。

当 *diag* 键选项不为键值对时, 则将其设置为主对角线元素。

这个矩阵命令自动更新 MaxMatrixCols 计数器, 即矩阵最大的列数。

见下方的几个例子。

例 6:

```
\diagonalmatrix[v,empty=\Box]{a,b,c,d}$ \
\diagonalmatrix[v,empty=\Box]{ 0={a,b,c,d} }$ \
\diagonalmatrix[v,empty=\Box]{auto-update, 2={a,b,c,d} }$ \
\diagonalmatrix[v,empty=\Box]{ 2={a,b,c,d} }$ \
\diagonalmatrix[v,empty=\Box]{auto-update, '0={a,b,c,d} }$ \
\diagonalmatrix[v,empty=\Box]{ '0={a,b,c,d} }$
```

$$\begin{vmatrix} a & \Box & \Box & \Box \\ \Box & b & \Box & \Box \\ \Box & \Box & c & \Box \\ \Box & \Box & \Box & d \end{vmatrix} \begin{vmatrix} a & \Box & \Box & \Box \\ \Box & b & \Box & \Box \\ \Box & \Box & c & \Box \\ \Box & \Box & \Box & d \end{vmatrix} \begin{vmatrix} \Box & \Box & a & \Box & \Box & \Box \\ \Box & \Box & \Box & b & \Box & \Box \\ \Box & \Box & \Box & \Box & c & \Box \\ \Box & \Box & \Box & \Box & \Box & d \end{vmatrix} \begin{vmatrix} \Box & \Box & \Box & a \\ \Box & \Box & b & \Box \\ \Box & c & \Box & \Box \\ d & \Box & \Box & \Box \end{vmatrix} \begin{vmatrix} \Box \end{vmatrix}$$

例 7:

```
\diagonalmatrix[v,empty=\Box]{ 0={a,b,c,d}, 1={h,j,j} }$ \
\diagonalmatrix[v,empty=\Box]{ 0={a,b,c,d}, 1={h,j,j,k} }$ \
\diagonalmatrix[v,empty=\Box]{ auto-update, 0={a,b,c,d}, 1={h,j,j,k} }$ \
\diagonalmatrix[v,empty=\Box]{ auto-update, 0={a,b,c,d}, 1={h,j,j,k}, -4={m} }$ \
```

$$\begin{vmatrix} a & h & \Box & \Box \\ \Box & b & j & \Box \\ \Box & \Box & c & j \\ \Box & \Box & \Box & d \end{vmatrix} \begin{vmatrix} a & h & \Box & \Box \\ \Box & b & j & \Box \\ \Box & \Box & c & j \\ \Box & \Box & \Box & d \end{vmatrix} \begin{vmatrix} a & h & \Box & \Box & \Box \\ \Box & b & j & \Box & \Box \\ \Box & \Box & c & j & \Box \\ \Box & \Box & \Box & d & k \end{vmatrix} \begin{vmatrix} a & h & \Box & \Box & \Box \\ \Box & b & j & \Box & \Box \\ \Box & \Box & c & j & \Box \\ \Box & \Box & \Box & d & k \\ m & \Box & \Box & \Box & \Box \end{vmatrix}$$

例 8:

```
\diagonalmatrix[v,empty=\Box]{ 0={1,2,3,4}, -1={11,22,33}, '-1={p,pp,ppp} }$ \
\diagonalmatrix[v,empty=\Box]{ 0={1,2,3,4}, 2=A, -1={\sum,\cup,\prod}, '-1={\int,\oint_a^n} }$ \
\diagonalmatrix[v,empty=\Box,sep=\C]{ 0=1 \C 2 \C 3 \C 4, 2=A, -1={\sum \C \cup \C \prod, '-1={\int \C \oint_a^n} }$ \

\diagonalmatrix=[v,empty=\Box,sep=\C]{ 0=1 \C 2 \C 3 \C 4, 2=A, -1={\sum \C \cup \C \prod, '-1={\int \C \oint_a^n} }$ \
{\ttfamily\meaning\physicxtmp}
```

$$\begin{vmatrix} 1 & \Box & \Box & \Box \\ 11 & 2 & \Box & p \\ \Box & 22 & pp & \Box \\ \Box & ppp & 33 & 4 \end{vmatrix} \begin{vmatrix} 1 & \Box & A & \Box \\ \sum & 2 & \Box & f \\ \Box & \cup & \oint_a^n & \Box \\ \Box & \Box & \prod & 4 \end{vmatrix} \begin{vmatrix} 1 & \Box & A & \Box \\ \sum & 2 & \Box & f \\ \Box & \cup & \oint_a^n & \Box \\ \Box & \Box & \prod & 4 \end{vmatrix}$$

macro:->1&\Box &A&\Box \|[0.0pt]\sum &2&\Box &\int \|[0.0pt]\Box &\cup &\oint_a^n&\Box \|[0.0pt]\Box &\Box &\prod &4

`\diagonalmatrix` 中可用的 `matrix` 类键值选项:

见表 2, 它们的具体用法见第 4.5 节。

`<diag` 键选项) 设置的即是 `diag` 键。

enhanced 模式下增加的可用键值:

表 2

expand	rows	cols	auto-update	empty
check	diag	diag+	diag-now	diag-data
item	item+	item-now	item-data	check-range
begin	end	args	args*	after-begin
after-begin+	after-end	after-end+	sepdim	type
saveto	saveto*	transpose	'	T
MaxMatrixCols	enhanced	!enhanced	sep	adi-order
beginning	beginning+	ending	ending+	

row-list	col-list	element-code	element-code*
element-except	element-except+	expand-element	row-iterate
col-iterate			

4.2 commamatrix

commamatrix, 即逗号分隔的矩阵。

```
\commamatrix    [matrix类键值选项] {<矩阵元素>}
\commamatrix =   [matrix类键值选项] {<矩阵元素>}
\commamatrix +   [matrix类键值选项] {<矩阵元素>}
\commamatrix = + [matrix类键值选项] {<矩阵元素>}
```

=, + 的功能同 \diagonalmatrix。

在非 enhanced 模式下, 做的工作仅仅是把 align 分隔符替换为 &, 把 cr 分隔符替换为 \\, 功能有限, 但速度更快。

这个矩阵命令 **不会** 自动更新 MaxMatrixCols 计数器, 需要通过 MaxMatrixCols 键设置, 或通过 \setcounter 设置。

见下方的几个例子。

例 9:

```
$\commamatrix[b]{a,b;c,d}$ \
$\commamatrix[b]{ {a,b} ; c,d}$ \
$\commamatrix[b]{a & b \\ c & d}$ \
$\commamatrix[b]{a, b & c \\ d & e, f}$ \
$\commamatrix[b, align=\T, cr=\N]{a \T b \T c \N d \T e \T f}$ \
$\commamatrix[b, align=\T, cr=\N]{a,a \T b;b \T c \N d \T e \T f}$ \

$\commamatrix=[b, align=\T, cr=\N]{a \T b \T c \N d \T e \T f}$ \
{\ttfamily\meaning\physicxtmp}
```

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} a, b \\ c & d \end{bmatrix} \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} a & b & c \\ d & e & f \end{bmatrix} \begin{bmatrix} a & b & c \\ d & e & f \end{bmatrix} \begin{bmatrix} a, a & b; b & c \\ d & e & f \end{bmatrix}$$

macro:->a &b &c \\[0.0pt]d &e &f

\commamatrix 可用的键值选项:

见表 3, 具体用法见第 4.5 节。

<矩阵元素> 设置的即是 array 键。

enhanced 模式下增加的键值选项:

表 3

array	expand	rows	cols	check-range
begin	end	args	args*	after-begin
after-end	sepdim	type	save-to	save-to*
MaxMatrixCols	enhanced	!enhanced	cr	align

row-list	col-list	element-code	element-code*
element-except	element-except+	expand-element	row-iterate
col-iterate			

4.3 qxmatrix

\qxmatrix 扩展的 \xmatrix。

\qxmatrix \qxmatrix [*matrix*类键值选项] {\main} {\last row} [\row] {\last col} [\col]
 \qxmatrix * [*matrix*类键值选项] {\main} {\last row} [\row] {\last col} [\col]
 \qxmatrix = [*matrix*类键值选项] {\main} {\last row} [\row] {\last col} [\col]
 \qxmatrix = * [*matrix*类键值选项] {\main} {\last row} [\row] {\last col} [\col]
 = 的作用同前所述。
 * 输出无穷维矩阵,即在最后的行和列添加 $\cdots; \ddots$ 。
 见如下几个例子。

例 10:

```
$\qxmatrix{A}{3}{3}$ \
$\qxmatrix{A}{7}{3}{9}{3}$ \
$\qxmatrix{A}{m}{2}{n}{2}$ \
```

```
$\qxmatrix[m]{A}{m}{2}{n}{2}$ \
$\qxmatrix*{A}{m}{2}{n}{2}$ \
$\qxmatrix[
  element-code=\displaystyle\pdv{\#1_{\#2}}{x_{\#3}},
  sepdim=2ex,
  p,
]{A}{m}{2}{n}{2}$ \
```

$$\begin{pmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{pmatrix} \begin{pmatrix} A_{11} & A_{12} & A_{13} & \cdots & A_{19} \\ A_{21} & A_{22} & A_{23} & \cdots & A_{29} \\ A_{31} & A_{32} & A_{33} & \cdots & A_{39} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ A_{71} & A_{72} & A_{73} & \cdots & A_{79} \end{pmatrix} \begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1n} \\ A_{21} & A_{22} & \cdots & A_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{m1} & A_{m2} & \cdots & A_{mn} \end{pmatrix}$$

$$\begin{matrix} A_{11} & A_{12} & \cdots & A_{1n} \\ A_{21} & A_{22} & \cdots & A_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{m1} & A_{m2} & \cdots & A_{mn} \end{matrix} \begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1n} & \cdots \\ A_{21} & A_{22} & \cdots & A_{2n} & \cdots \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ A_{m1} & A_{m2} & \cdots & A_{mn} & \cdots \\ \vdots & \vdots & & \vdots & \ddots \end{pmatrix} \begin{pmatrix} \frac{\partial A_1}{\partial x_1} & \frac{\partial A_1}{\partial x_2} & \cdots & \frac{\partial A_1}{\partial x_n} \\ \frac{\partial A_2}{\partial x_1} & \frac{\partial A_2}{\partial x_2} & \cdots & \frac{\partial A_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial A_m}{\partial x_1} & \frac{\partial A_m}{\partial x_2} & \cdots & \frac{\partial A_m}{\partial x_n} \end{pmatrix}$$

例 11:

```
\newcommand {\twotwomat}[2] [] {\qmatrix[type=#2,#1]{A}{n}[2]{n}[2]}
\setmatrixtype*{jacobian}{
  element-code=\displaystyle\pdv{##1_{{##2}}}{x_{{##3}}},
  sepdim=2ex}
$\twotwomat[p]{jacobian}$

\setmatrixtype*{parenmat}{ element-code=##1_{{##2}}(x_{{##3}}) }
$\twotwomat{parenmat}$

\setmatrixtype*{plusmain}{ element-code={
  \ifthenelse{\equal{##2}{##3}}{1+}{ }##1_{{##2}}(x_{{##3}})
} }
$\twotwomat[v]{plusmain}$
```

$$\left(\begin{array}{cccc} \frac{\partial A_1}{\partial x_1} & \frac{\partial A_1}{\partial x_2} & \cdots & \frac{\partial A_1}{\partial x_n} \\ \frac{\partial A_2}{\partial x_1} & \frac{\partial A_2}{\partial x_2} & \cdots & \frac{\partial A_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial A_n}{\partial x_1} & \frac{\partial A_n}{\partial x_2} & \cdots & \frac{\partial A_n}{\partial x_n} \end{array} \right) \begin{array}{cccc} A_1(x_1) & A_1(x_2) & \cdots & A_1(x_n) \\ A_2(x_1) & A_2(x_2) & \cdots & A_2(x_n) \\ \vdots & \vdots & \ddots & \vdots \\ A_n(x_1) & A_n(x_2) & \cdots & A_n(x_n) \end{array}$$

$$\left| \begin{array}{cccc} 1 + A_1(x_1) & A_1(x_2) & \cdots & A_1(x_n) \\ A_2(x_1) & 1 + A_2(x_2) & \cdots & A_2(x_n) \\ \vdots & \vdots & \ddots & \vdots \\ A_n(x_1) & A_n(x_2) & \cdots & 1 + A_n(x_n) \end{array} \right|$$

一个稍复杂的例子:

例 12:

```
% \usepackage{ifthen}
\def \hessianaux#1{
  \def \twopartital##1##2##3{%
    \displaystyle\ifthenelse{\equal{##2}{##3}}
      {\pdv[2]{##1}{#1_{{##2}}}}
      {\pdv{##1}{#1_{{##2}}}{#1_{{##3}}}}}
\newcommand{\hessian}[4] [] {%
  \hessianaux{#3}%
  \qmatrix[element-code=twopartital,#1]{#2}{#4}[1]{#4}[1]}
$\hessian[v,sepdim=1ex]{f}{x}{m}$
$\hessian[b,sepdim=1ex]{\mathcal{H}}{p}{n}$
```

$$\left| \begin{array}{ccc} \frac{\partial^2 f}{\partial x_1^2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_m \partial x_1} & \cdots & \frac{\partial^2 f}{\partial x_m^2} \end{array} \right| \left[\begin{array}{ccc} \frac{\partial^2 \mathcal{H}}{\partial p_1^2} & \cdots & \frac{\partial^2 \mathcal{H}}{\partial p_1 \partial p_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 \mathcal{H}}{\partial p_n \partial p_1} & \cdots & \frac{\partial^2 \mathcal{H}}{\partial p_n^2} \end{array} \right]$$

`\qmatrix` 可用的键值选项:

见表 4, 具体用法见第 4.5 节。

表 4

array	expand	rows	cols
auto-update	main	row-list	col-list
infinite	!infinite	element-code	element-code*
element-except	element-except+	expand-element	row-iterate
col-iterate	begin	end	args
args*	sepdim	type	saveto
saveto*	MaxMatrixCols		

4.4 genegralmatrix

generalmatrix, 即通用矩阵命令。

```

\generalmatrix \generalmatrix + {<matrix类键值选项>}
\generalmatrix = + {<matrix类键值选项>}
\generalmatrix + * {<matrix类键值选项>}
\generalmatrix = + * {<matrix类键值选项>}
\generalmatrix {<matrix类键值选项>} {<main>} {<last row>} [<row>] {<last col>} [<col>]
\generalmatrix = {<matrix类键值选项>} {<main>} {<last row>} [<row>] {<last col>} [<col>]
\generalmatrix * {<matrix类键值选项>} {<main>} {<last row>} [<row>] {<last col>} [<col>]
\generalmatrix = * {<matrix类键值选项>} {<main>} {<last row>} [<row>] {<last col>} [<col>]

```

不带 + 的命令就是 \qxmatrix 命令, 不过其第一个参数是必须用 { } 给定的。

这个矩阵命令自动更新 MaxMatrixCols 计数器, 即矩阵最大的列数。

4.5 matrix 类通用键值选项

array

expand expand = {none|text-expand|f|romanual|x|edef}

初始值 = none

对输入的元素进行扩展, 即先对 array、diag、item 扩展, 再进行处理。

rows rows = {整数}

cols cols = {整数}

auto-update auto-update = true|false

初始值 = false

是否自动更新行数和列数。

main

row-list

col-list

设置相应行、列的角标。

infinite

!infinite

是否无穷维矩阵。

element-code element-code = {<code>}

element-code* element-code* = {except-empty|except-dots|except-tl|except-regex|only-regex|<macro name>}

element-code 使用三个参数, 分别为 main、行角标、列角标。

element-except

element-except+

<hr/> <hr/> expand-element <hr/>	expand-element = true false	初始值 = false
	是否对生成的矩阵值进行扩展, 一般情况是不需要的, 当使用 <code>saveto</code> 保存到宏中, 则可能需要启用该选项。	
<hr/> empty <hr/>	当矩阵的某元素未被设置时, 使用其值设置之。	
<hr/> check <hr/>	check = \langle none empty ignore igep all \rangle	
	是否对输入的元素进行检查。	
	empty 选项检查当输入的元素为空时, 使用 empty 键的值替换之。	
	ignore 检查当输入的元素为 <code>\PHYSICXIGNORE</code> 时, 不对该元素进行设置操作。	
	igep 对 empty 和 ignore 进行检查。	
<hr/> row-iterate <hr/> col-iterate <hr/>	设置行、列的角标的索引方法, 它们接收一个参数, 分别代表对应元素的行索引和列索引。	
<hr/> last-row <hr/> last-col <hr/>		
<hr/> diag <hr/> diag+ <hr/> diag-now <hr/> diag-data <hr/> diag-data+ <hr/>	diag = $\{\langle$ diag 键选项 $\rangle\}$ diag+ = $\{\langle$ diag 键选项 $\rangle\}$ diag-now = $\{\langle$ diag 键选项 $\rangle\}$ diag-data = $\{\langle$ 预先定义的diag数据 $\rangle\}$ diag-data+ = $\{\langle$ 预先定义的diag数据 $\rangle\}$	
	diag-now 立即解析给出的 \langle diag 键选项 \rangle , 不待 adi 处理, 或 adi 处理完毕后还需设置矩阵元素。可用于 beginning、ending 键中进行额外的设置。	
	diag-data 可使用 <code>\setmatrixxdata</code> 设置的数据, 见第 5 节。	
<hr/> item <hr/> item+ <hr/> item-now <hr/> item-data <hr/> item-data+ <hr/>		
<hr/> check-range <hr/>	check-range = true false	初始值 = true
	当解析 item 选项时, 是否检查元素索引越界, 为真时, 不设置越界元素。	
<hr/> begin <hr/> end <hr/>	begin = $\{\langle$ code $\rangle\}$ end = $\{\langle$ code $\rangle\}$	
	设置输出矩阵时使用的环境。一般为 <code>begin=\begin{matrix}, end=\end{matrix}</code> 。	
	但是可以是任意代码, 一般情况下 begin 和 end 需配对。	
<hr/> args <hr/> args* <hr/>	args = \langle code \rangle args* = \langle code \rangle	
	当 begin 键使用的环境需要参数时, 可使用这个键设置。	
	args 设置方括号括起来的参数, 将 \langle code \rangle 放到 begin 键的代码之后。	
	args* 将 \langle code \rangle 放到 begin 键的代码之后。	
	如: <code>begin=\begin{matrix*}, args=c</code> , 则为 <code>\begin{matrix*}[c]</code> 。	
	<code>begin=\begin{array}, args*={[t]{ccc}}</code> , 则为 <code>\begin{array}[t]{ccc}</code> 。	
<hr/> after-begin <hr/> after-begin+ <hr/> after-end <hr/> after-end+ <hr/>		

<hr/> sepdim <hr/>	sepdim = $\langle dim \rangle$	初始值 = <code>Opt</code>
<hr/> type <hr/>	type = $\langle m p b B v V sm ... \rangle$	初始值 = <code>m</code>
设置输出矩阵的类型。 <code>m</code> 使用 <code>matrix</code> 环境, <code>p</code> 使用 <code>pmatrix</code> 环境, 等。		
当加载了 <code>mathtools</code> 宏包或使用了 <code>mathtools</code> 宏包选项时, 还额外定义了 <code>m*</code> 和 <code>sm*</code> 、 <code>sp*</code> 、 <code>sp*</code> 等环境, 它们代表 <code>martix*</code> 、 <code>smallmatrix*</code> 、 <code>psmallmatrix</code> 等环境。		
还可使用由 <code>\setmatrixtype</code> 定义的类型。见第 5 节。		
一般情况下, 如果 <code>type</code> 的值与 <code>matrix</code> 类的键没有相同的名字, 则可省略 <code>type</code> , 直接写它的值。不能保证未来版本不会增加新的键, 若要使得简写仍然有效, 应该保证它的名字比较特别, 使得出现相同键名的概率低。		
<hr/> saveto <hr/> saveto*	saveto = $\langle macro\ name \rangle$ saveto* = $\langle macro\ name \rangle$	
将矩阵的值保存到相应的宏中。		
<hr/> transpose, ' T <hr/>	transpose = <code>true false</code>	初始值 = <code>false</code>
为真, 则将矩阵元素转置, 不对角标转置。		
<hr/> MaxMatrixCols <hr/>	MaxMatrixCols = $\langle 整数 \rangle$	
设置 <code>MaxMatrixCols</code> 计数器的值。一般情况下无需给出键名, 直接使用 $\langle 整数 \rangle$ 即可。		
<hr/> enhanced <hr/>	enhanced = <code>true false</code> !enhanced	初始值 = <code>false</code>
是否启用 <code>enhanced</code> 模式。		
<hr/> cr <hr/> align <hr/> sep <hr/>	cr = $\langle 符号 \rangle$ align = $\langle 符号 \rangle$ sep = $\langle 符号 \rangle$	初始值 = <code>;</code> 初始值 = <code>,</code> 初始值 = <code>,</code>
设置分隔符。 $\langle 符号 \rangle$ 可以是任意记号, 不会对其进行展开。		
sep 设置 <code>diag</code> 、 <code>row-list</code> 、 <code>col-list</code> 的分隔符。		
cr 设置换行 (<code>\</code>) 对应的符号。align 设置 <code>align (&)</code> 对应的符号, 用于 <code>array</code> 键或 <code>\commamatrix</code> 中。		
<hr/> adi-order <hr/>	adi-order = $\langle adi dia iad aid ida dai \rangle$	初始值 = <code>adi</code>
设置 <code>array/main</code> 、 <code>diag</code> 、 <code>item</code> 的处理顺序。		
<hr/> beginning <hr/> beginning+ ending ending+ <hr/>	设置在 <code>adi</code> 处理前后要执行的代码, 可以是任意代码, 一般用于设置。	

4.6 例子

例 13:

```

 $\commamatrix[
  begin=\begin{array}, end=\end{array}, args*={{\clr}}, \% two {{ }}
]{A, E, I; MNOP, QRST, UVWX}$
 $\commamatrix[
  begin=\left[\begin{array}, end=\end{array}\right], args*={{\clr}},
]{A, E, I; MNOP, QRST, UVWX}$$$ 
```

$$\begin{array}{ccc} A & E & I \end{array} \left[\begin{array}{ccc} A & E & I \\ MNOP & QRST & UVWX \end{array} \right]$$

例 14:

```

 $\backslash$ generalmatrix+{rows=5,cols=4,main=A}$
 $\backslash$ generalmatrix+{array={ 1,2,3,4; a,b,c,d; \oplus,\otimes,\cup,\spadesuit
    },rows=3,cols=4,type=p}$
 $\backslash$ generalmatrix+{
    array={ 1,2,3,4; a,b,c,d; \oplus,\otimes,\cup,\spadesuit },
    type=p,auto-update
}$

```

$$\begin{array}{cccc}
 A_{11} & A_{12} & A_{13} & A_{14} \\
 A_{21} & A_{22} & A_{23} & A_{24} \\
 A_{31} & A_{32} & A_{33} & A_{34} \\
 A_{41} & A_{42} & A_{43} & A_{44} \\
 A_{51} & A_{52} & A_{53} & A_{54}
 \end{array}
 \begin{pmatrix}
 1_{11} & 2_{12} & 3_{13} & 4_{14} \\
 a_{21} & b_{22} & c_{23} & d_{24} \\
 \oplus_{31} & \otimes_{32} & \cup_{33} & \spadesuit_{34}
 \end{pmatrix}
 \begin{pmatrix}
 1_{11} & 2_{12} & 3_{13} & 4_{14} \\
 a_{21} & b_{22} & c_{23} & d_{24} \\
 \oplus_{31} & \otimes_{32} & \cup_{33} & \spadesuit_{34}
 \end{pmatrix}$$

例 15:

```

 $\backslash$ generalmatrix + {
    array={ 1,2,3,4; a,b,c,d; \oplus,\otimes,\cup,\spadesuit },
    type=v,rows=3,cols=4,
    row-list={A,B,C}, col-list={\blacklozenge,\Box,\heartsuit}
}$
 $\backslash$ generalmatrix + * {
    array={ 1,2,3,4; a,b,c,d; \oplus,\otimes,\cup,\spadesuit },
    type=v,rows=3,cols=4,
    row-list={A,B,C}, col-list={\blacklozenge,\Box,\heartsuit}
}$
 $\backslash$ generalmatrix + * {
    array={ 1,2,3,4; a,b,c,d; \oplus,\otimes,\cup,\spadesuit },
    type=v,rows=3,cols=4, element-code*=except-dots,
    row-list={A,B,C}, col-list={\blacklozenge,\Box,\heartsuit}
}$

```

$$\begin{array}{ccccc}
 \left| \begin{array}{ccccc}
 1_{A\blacklozenge} & 2_{A\Box} & 3_{A\heartsuit} & 4_A & \cdots_A \\
 a_{B\blacklozenge} & b_{B\Box} & c_{B\heartsuit} & d_B & \cdots_B \\
 \oplus_{C\blacklozenge} & \otimes_{C\Box} & \cup_{C\heartsuit} & \spadesuit_C & \cdots_C \\
 \vdots_{\blacklozenge} & \vdots_{\Box} & \vdots_{\heartsuit} & \vdots & \ddots
 \end{array} \right| &
 \left| \begin{array}{ccccc}
 1_{A\blacklozenge} & 2_{A\Box} & 3_{A\heartsuit} & 4_A & \cdots_A \\
 a_{B\blacklozenge} & b_{B\Box} & c_{B\heartsuit} & d_B & \cdots_B \\
 \oplus_{C\blacklozenge} & \otimes_{C\Box} & \cup_{C\heartsuit} & \spadesuit_C & \cdots_C \\
 \vdots_{\blacklozenge} & \vdots_{\Box} & \vdots_{\heartsuit} & \vdots & \ddots
 \end{array} \right| &
 \left| \begin{array}{ccccc}
 1_{A\blacklozenge} & 2_{A\Box} & 3_{A\heartsuit} & 4_A & \cdots_A \\
 a_{B\blacklozenge} & b_{B\Box} & c_{B\heartsuit} & d_B & \cdots_B \\
 \oplus_{C\blacklozenge} & \otimes_{C\Box} & \cup_{C\heartsuit} & \spadesuit_C & \cdots_C \\
 \vdots_{\blacklozenge} & \vdots_{\Box} & \vdots_{\heartsuit} & \vdots & \ddots
 \end{array} \right|
 \end{array}$$

例 16:

```

$\generalmatrix + {
  rows=5,cols=5,type=v,
  main=A, diag={ 0={1,2,3,4,5,6}, '0={-1,-2,-3,-4,-5,-6} },
}$
$\generalmatrix + {
  rows=5,cols=5,type=v, element-code={#1},
  main=A, diag={ 0={1,2,3,4,5,6}, '0={-1,-2,-3,-4,-5,-6} },
}$
$\generalmatrix + {
  rows=5,cols=5,type=v, element-code={#1},
  main=A, diag={ 0={1,2,3,4,5,6}, '0={-1,-2,-3,-4,-5,-6} },
  item={ {1,3}{2,4}=\Box },
}$

```

$$\left| \begin{array}{ccccc} 1_{11} & A_{12} & A_{13} & A_{14} & -1_{15} \\ A_{21} & 2_{22} & A_{23} & -2_{24} & A_{25} \\ A_{31} & A_{32} & -3_{33} & A_{34} & A_{35} \\ A_{41} & -4_{42} & A_{43} & 4_{44} & A_{45} \\ -5_{51} & A_{52} & A_{53} & A_{54} & 5_{55} \end{array} \right| \left| \begin{array}{ccccc} 1 & A & A & A & -1 \\ A & 2 & A & -2 & A \\ A & A & -3 & A & A \\ A & -4 & A & 4 & A \\ -5 & A & A & A & 5 \end{array} \right| \left| \begin{array}{ccccc} 1 & \square & A & \square & -1 \\ A & 2 & A & -2 & A \\ A & \square & -3 & \square & A \\ A & -4 & A & 4 & A \\ -5 & A & A & A & 5 \end{array} \right|$$

例 17:

```

% new a \generalmatrix command, use enhanced mode, with 1 arg
\newgeneralmatrix + \ffempty [1] { rows=5,cols=5,element-code={##1},type=v, #1 }
$\ffempty{ main=A, diag={ 0={1,2,3,4,5,6}, '0={-1,-2,-3,-4,-5,-6} } }$
$\ffempty{
  main=A, diag={ 0={1,2,3,4,5,6}, '0={-1,-2,-3,-4,-5,-6} },
  item={ {1,3}{2,4}=\Box }
}$
$\ffempty{
  main=A, diag={ 0={1,2,3,4,5,6}, '0={-1,-2,-3,-4,-5,-6} },
  item={
    {1,3}{2,4}=\Box, % (1,2), (1,4), (3,2), (3,4) = \Box
    {4}{5}=\PHYSICXIGNORE, % ignore setting of (4,5)
    {-}{5}=\blacksquare, % col=5
    {4}{-}=\heartsuit, % row=4
  }
}$

```

$$\left| \begin{array}{ccccc} 1 & A & A & A & -1 \\ A & 2 & A & -2 & A \\ A & A & -3 & A & A \\ A & -4 & A & 4 & A \\ -5 & A & A & A & 5 \end{array} \right| \left| \begin{array}{ccccc} 1 & \square & A & \square & -1 \\ A & 2 & A & -2 & A \\ A & \square & -3 & \square & A \\ A & -4 & A & 4 & A \\ -5 & A & A & A & 5 \end{array} \right| \left| \begin{array}{ccccc} 1 & \square & A & \square & \blacksquare \\ A & 2 & A & -2 & \blacksquare \\ A & \square & -3 & \square & \blacksquare \\ \heartsuit & \heartsuit & \heartsuit & \heartsuit & A \\ -5 & A & A & A & \blacksquare \end{array} \right|$$

例 18:

```
% xparse-like new command
\NewGeneralMatrix + \fftest { D(){adi} 0{ } } {
  rows=5,cols=5, element-code={##1},
  main=A, diag={ 0={1,2,3,4,5,6}, '0={-1,-2,-3,-4,-5,-6} },
  item={
    {1,3}{2,4}=\Box, % (1,2), (1,4), (3,2), (3,4) = \Box
    {4}{5}=\PHYSICXIGNORE, % ignore setting of (4,5)
    {-}{5}=\blacksquare, % col=5
    {4}{-}=\heartsuit, % row=4
  }, adi-order=#1, #2 }

$\fftest[type=v]$ % adi-order=adi, process order: array/main, diag, item
$\fftest(adi)[type=v]$ % adi-order=aid, process order: array/main, item, diag
$\fftest(adi)[type=v,
  diag+={ 2={HH,II,JJ}, -2={ , , } }, % add to diag
  item+={ {4}{1}={ } }, % add to item
  ending=\physicxset*{matrix}{ item-now={ {4}{4}=\fbox{LL} } }, % set item at ending
]$

$\fftest(adi)[type=v,
  diag+={ 2={HH,II,JJ}, -2={ , , } },
  item+={ {4}{1}={ } },
  ending=\physicxset*{matrix}{ item-now={ {4}{4}=\fbox{LL} } },
  saveto=\savetosomething,
]$
{\ttfamily\meaning\savetosomething}
```

$$\left| \begin{array}{ccccc} 1 & \square & A & \square & \blacksquare \\ A & 2 & A & -2 & \blacksquare \\ A & \square & -3 & \square & \blacksquare \\ \heartsuit & \heartsuit & \heartsuit & \heartsuit & A \\ -5 & A & A & A & \blacksquare \end{array} \right| \left| \begin{array}{ccccc} 1 & \square & A & \square & -1 \\ A & 2 & A & -2 & \blacksquare \\ A & \square & -3 & \square & \blacksquare \\ \heartsuit & -4 & \heartsuit & 4 & A \\ -5 & A & A & A & 5 \end{array} \right| \left| \begin{array}{ccccc} 1 & \square & HH & \square & -1 \\ A & 2 & A & II & \blacksquare \\ & \square & -3 & \square & JJ \\ & & \heartsuit & \boxed{LL} & A \\ -5 & A & & A & 5 \end{array} \right|$$

macro:->1&\Box &HH&\Box &-1\\[0.0pt]A&2&A&II&\blacksquare \\[0.0pt]&\Box &-3&\Box &JJ\\[0.0pt] &&\heartsuit &\fbox{LL}&A\\[0.0pt]-5&A&A&A&5

第 5 节 杂项

`\physicxset` `\physicxset {<通用键值选项>}`
`\physicxset * {<类型>} {<类型可用的键值选项>}`

`physicx` 宏包的设置命令。

`<类型>` 为 `quantity`、`matrix` 等。

`\setquantitytype` `\setquantitytype {<type name>} {<quantity类键选项>}`

自定义 `quantity` 类的 `type`。

`\physicxtmp` `\physicxempty` 保存 `empty` 键的值。`\physicxexcept` 保存 `element-except` 的值。
`\physicxempty` 永远不要在正文中使用 `\PHYSICXIGNORE`。仅应 `diag`、`item` 键的设置中使用。
`\physicxexcept`
`\PHYSICXIGNORE`

<code>\setmatrixtype</code>	<code>\setmatrixtype {<type name>} {<begin>} {<end>}</code>
<code>\setmatrixdata</code>	<code>\setmatrixtype * {<type name>} {<matrix类键选项>}</code> <code>\setmatrixdata {<data type>} {<data name>} {<data>}</code>

`<data type>` 为 `item` 或 `diag`, 将 `<data>` 保存, 以后可以在 `item-data`、`diag-data` 中通过 `<data name>` 来引用。

5.1 定义新的括号命令

<code>\newxquantity</code>	<code>\newxquantity \<cmd> [{<arg nums>}] [{<default>}] {<quantity keys>} {<code>}</code>
<code>\NewXQuantity</code>	<code>\NewXQuantity \<cmd> {<arg spec>} {<quantity keys>} {<code>}</code>

定义新的矩阵命令。`\newxquantity` 与 $\text{\LaTeX} 2_{\epsilon}$ 的 `\newcommand` 形式相同。`\NewXQuantity` 与 `xparse` 的 `\NewDocumentCommand` 形式相同。

其中 `\NewXQuantity` 命令使用的宏变量从第 2 开始, 第 1 个保留作内部使用。

可以使用 `\newcommand` 或 `\NewDocumentCommand` 配合 `\xquantity` 命令来定义新的括号命令。

实际上, `\qxqty` 的定义就是

例 19:

```
\NewXQuantity \qxqty { 0{ } m } {#2} {#3}
```

例 20:

```
\newxquantity \xcommu [3] [] {#1, type=b} {#2,#3}
$ \xcommu{A}{B} \quad \xcommu{\dfrac{A}{B}}{\dfrac{C}{D}} $ \quad
$ \xcommu[\Big]{A}{B} \quad \quad \quad \xcommu[\big]{\dfrac{A}{B}}{\dfrac{C}{D}} $
```

$$[A, B] \quad \left[\frac{A}{B}, \frac{C}{D} \right] \quad [A, B] \quad \left[\frac{A}{B}, \frac{C}{D} \right]$$

<code>\@declarequantitycmd</code>	<code>\@declarequantitycmd 0/1 0/1 \<cmd> {<code>}</code>
<code>\@declareparencmd</code>	<code>\@declareparencmd \<cmd> {<arg spec>} {<replace>} {<pre code>} <left> <right> {<post>}</code>

这两个命令不宜再使用。

由 `\@declarequantitycmd` 命令定义的 `quantity` 命令, 其后除可使用 `\big`、`\Big` 等命令外, 也支持任意命令。宏变量从第 4 个开始, 前 3 个保留作内部使用。

由 `\@declareparencmd` 命令定义的 `quantity` 命令, 其后仅可使用 `\big`、`\Big`、`\bigg`、`\Bigg` 这四个命令。宏变量从第 6 个开始, 前 5 个保留作内部使用。

几个例子:

例 21:

```
% \catcode`\ =9 \catcode`\@=11
\@declarequantitycmd 1 1 \quantity
{
  { !g } { { \{ } { #4 } { \} } }
  { !o } { { [ } { #5 } { ] } }
  { !d() } { { ( } { #6 } { ) } }
  { !d|| } { { \vert } { #7 } { \vert } }
  { !d<> } { { \langle } { #8 } { \rangle } }
  { !d== } { { \Vert } { #9 } { \Vert } }
}
\@declarequantitycmd 1 0 \matrixquantity
{
  { !g }
  {
    { \IfBooleanT{#3}{\left\{ } }
    { \begin{matrix} #4 \end{matrix} }
    { \IfBooleanT{#3}{\right\} }
  }
  { !o } { { \begin{bmatrix} } { #5 } { \end{bmatrix} } }
  { !d() }
  {
    { \IfBooleanTF{#3}{\left\lgroup}{\left( } }
    { \begin{matrix} #6 \end{matrix} }
    { \IfBooleanTF{#3}{\right\rgroup}{\right)} }
  }
  { !d|| } { { \begin{vmatrix} } { #7 } { \end{vmatrix} } }
  { !d<> } { { \left\langle } { \begin{matrix} #8 \end{matrix} } { \right\rangle } }
  { !d== } { { \begin{Vmatrix} } { #9 } { \end{Vmatrix} } }
}
}
```

例 22:

```
% \catcode`\ =9 \catcode`\@=11
\@declareparencmd \pqty { m } { #6 } { } { ( ) { } }
\@declareparencmd \absolutevalue { m } { #6 } { } { \vert \vert { } }
\@declareparencmd \OOrder { m } { #6 } { } { \mathcal{0} } { ( ) { } }
\@declareparencmd \commutator { m m } { #6 , #7 } { } { [ ] { } }
```

5.2 定义新的矩阵命令

<code>\newdiagonalmatrix</code>	<code>\newdiagonalmatrix</code>	<code>\langle cmd \rangle</code>	<code>[(arg nums)]</code>	<code>[(default)]</code>	<code>{\langle matrix keys \rangle}</code>	<code>{\langle diag keys \rangle}</code>
<code>\NewDiagonalMatrix</code>	<code>\newdiagonalmatrix +</code>	<code>\langle cmd \rangle</code>	<code>[(arg nums)]</code>	<code>[(default)]</code>	<code>{\langle matrix keys \rangle}</code>	<code>{\langle diag keys \rangle}</code>
<code>\newcommamatrix</code>	<code>\NewDiagonalMatrix</code>	<code>\langle cmd \rangle</code>	<code>{\langle args spec \rangle}</code>		<code>{\langle matrix keys \rangle}</code>	<code>{\langle diag keys \rangle}</code>
<code>\NewCommaMatrix</code>	<code>\NewDiagonalMatrix +</code>	<code>\langle cmd \rangle</code>	<code>{\langle args spec \rangle}</code>		<code>{\langle matrix keys \rangle}</code>	<code>{\langle diag keys \rangle}</code>
<code>\newgeneralmatrix</code>	<code>\newcommamatrix</code>	<code>\langle cmd \rangle</code>	<code>[(arg nums)]</code>	<code>[(default)]</code>	<code>{\langle matrix keys \rangle}</code>	<code>{\langle comma value \rangle}</code>
<code>\NewGeneralMatrix</code>	<code>\newcommamatrix +</code>	<code>\langle cmd \rangle</code>	<code>[(arg nums)]</code>	<code>[(default)]</code>	<code>{\langle matrix keys \rangle}</code>	<code>{\langle comma value \rangle}</code>
	<code>\NewCommaMatrix</code>	<code>\langle cmd \rangle</code>	<code>{\langle args spec \rangle}</code>		<code>{\langle matrix keys \rangle}</code>	<code>{\langle comma value \rangle}</code>
	<code>\NewCommaMatrix +</code>	<code>\langle cmd \rangle</code>	<code>{\langle args spec \rangle}</code>		<code>{\langle matrix keys \rangle}</code>	<code>{\langle comma value \rangle}</code>
	<code>\newgeneralmatrix</code>	<code>\langle cmd \rangle</code>	<code>[(arg nums)]</code>	<code>[(default)]</code>	<code>{\langle matrix keys \rangle}</code>	
	<code>\newgeneralmatrix +</code>	<code>\langle cmd \rangle</code>	<code>[(arg nums)]</code>	<code>[(default)]</code>	<code>{\langle matrix keys \rangle}</code>	
	<code>\NewGeneralMatrix</code>	<code>\langle cmd \rangle</code>	<code>{\langle args spec \rangle}</code>		<code>{\langle matrix keys \rangle}</code>	
	<code>\NewGeneralMatrix +</code>	<code>\langle cmd \rangle</code>	<code>{\langle args spec \rangle}</code>		<code>{\langle matrix keys \rangle}</code>	

例 23:

```

\newdiagonalmatrix \xdmat [2] [] {#1} {0={#2}}
$ \xdmat{1,2,3,4}$ \quad $\xdmat[v]{1,2,3,4}$ \quad $\xdmat[v,empty=0]{1,2,3,4} $
\quad
$ \xdmat{1, \commamatrix{2&3\\4&5}} $ \quad
$ \xdmat{1, \mqty{2&3\\4&5}} $ \quad

\newdiagonalmatrix \xxdmat [3] [] {#1, diag={true, 0={#2}, #3} } {}
\newdiagonalmatrix \xadmat [2] [] {#1} {auto-update,'0={#2}}
$ \xxdmat{1,2,3,4}-{0={a,b,c,d}} $ \quad
$ \xxdmat[b]{1,2,5}-{'-1={3,4}} $ \quad
$ \xadmat{a,b,c,d} $

```

$$\begin{array}{c}
 \begin{array}{cccc}
 1 & & & \\
 & 2 & & \\
 & & 3 & \\
 & & & 4
 \end{array}
 \left| \begin{array}{cccc}
 1 & & & \\
 & 2 & & \\
 & & 3 & \\
 & & & 4
 \end{array} \right|
 \left| \begin{array}{cccc}
 1 & 0 & 0 & 0 \\
 0 & 2 & 0 & 0 \\
 0 & 0 & 3 & 0 \\
 0 & 0 & 0 & 4
 \end{array} \right|
 \begin{array}{cccc}
 1 & & & \\
 & 2 & 3 & \\
 & & & 2 & 3 \\
 & & & 4 & 5
 \end{array}
 \end{array}$$

$$\begin{array}{c}
 \begin{array}{cccc}
 1 & & a & \\
 & 2 & b & \\
 & c & 3 & \\
 d & & & 4
 \end{array}
 \left[\begin{array}{cc}
 1 & \\
 & 2 & 3 \\
 & & 4 & 5
 \end{array} \right]
 \begin{array}{cccc}
 & & a & \\
 & & b & \\
 & c & & \\
 d & & &
 \end{array}
 \end{array}$$

`\newgeneralmatrix`、`\NewGeneralMatrix` 的例子见例 17 和例 18。

版本历史

v0.1.2	(2021/11/11)	v0.2	(2021/11/12)
General: 新增 quantity 选项。	1	General: 新增 \xquantity、\qxqty 命令。	1
v0.1.3	(2021/11/12)	移除 \Order 命令。	3
General: 将 quantity 重命名为 reqty。	1		

代码索引

意大利体的数字表示描述对应索引项的页码;带下划线的数字表示定义对应索引项的代码行号;罗马字体的数字表示使用对应索引项的代码行号。

Symbols			
!infinite	11	diag	12
'	13	diag+	12
'-1	6	diag-data	12
'1	6	diag-data+	12
-1	6	diag-now	12
.	6	\diagonalmatrix	5-8
\(cmd)	17, 18	E	
\(macro name)	13	element-code	11
Numbers		element-code*	11
0	6	element-except	11
1	6	element-except+	11
A		empty	12
adi-order	13	end	12
after-begin	12	ending	13
after-begin+	12	ending+	13
after-end	12	enhanced	13
after-end+	12	expand	11
align	13	expand-element	12
args	2, 12	F	
args*	2, 12	false	6
array	11	G	
auto-update	6, 11	\genegralmatrix	5
B		\generalmatrix	6, 11
begin	12	I	
beginning	13	infinite	11
beginning+	13	item	12
C		item+	12
check	12	item-data	12
check-range	12	item-data+	12
code	2	item-now	12
col-iterate	12	L	
col-list	11	last-col	12
cols	11	last-row	12
\commamatrix	5, 8, 13	left	2
cr	13	left-size	2
D		M	
\dd	1	main	11

\matrixquantity	4	right-size	2
MaxMatrixCols	13	row-iterate	12
\mqty	4	row-list	11
		rows	11
N		S	
\NewCommaMatrix	18	saveto	13
\newcommamatrix	18	saveto*	13
\NewDiagonalMatrix	18	sep	13
\newdiagonalmatrix	18	sepdim	13
\NewDocumentCommand	17	\setmatrixdata	12,17
\NewGeneralMatrix	18,19	\setmatrixtype	13,17
\newgeneralmatrix	18,19	\setquantitytype	16
\NewXQuantity	17	size	2
\newxquantity	17	\smallmatrixquantity	4
noauto	2	\smqty	4
noauto-update	6		
O		T	
\OOrder	3	T	13
\oorder	3	TeX and L ^A T _E X ₂ _ε commands:	
\order	3	\@declareparencmd	17
		\@declarequantitycmd	17
P		\Big	17
\physicxempty	16	\big	17
\physicxexcept	16	\Bigg	17
\PHYSICXIGNORE	12,16	\bigg	17
\physicxset	16	\left	2,3
\physicxtmp	6,16	\newcommand	17
post	2	\right	2,3
pre	2	\setcounter	8
Q		transpose	13
\quantity	4	true	6
\qxmatrix	5,9–11	type	3,13
\qxqty	1,17		
R		X	
right	2	\xmatrix	9
		\xquantity	1,17