

physicx, 扩展的 physics 宏包

雾月, Longaster*

2021 年 11 月 6 日 v0.1

目录

第 1 节 简介	1	4.3 qxmatrix	7
第 2 节 基本用法与宏包选项	1	4.4 genegralmatrix	9
第 3 节 括号, quantity	1	4.5 matrix 类通用键值选项	9
第 4 节 矩阵, matrix	3	4.6 例子	11
4.1 diagonalmatrix	4	第 5 节 杂项	14
4.2 commamatrix	6	5.1 定义新的括号命令	15
		5.2 定义新的矩阵命令	16

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, 不准备兼容低版本, 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。

*Email: longaster@163.com

第 3 节 括号, *quantity*

对括号类命令的修改并不多,主要增加了 `\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`、`\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>	$\backslash\text{qty}(\backslash\text{typical})_{\square} \rightarrow (\blacksquare)$ $\backslash\text{qty}(\backslash\text{tall})_{\square} \rightarrow (\blacksquare)$ $\backslash\text{qty}(\backslash\text{grande})_{\square} \rightarrow (\blacksquare)$ $\backslash\text{qty}[\backslash\text{typical}]_{\square} \rightarrow [\blacksquare]$ $\backslash\text{qty} \backslash\text{typical} _{\square} \rightarrow \blacksquare $ $\backslash\text{qty}\{\backslash\text{typical}\}_{\square} \rightarrow \{\blacksquare\}$ $\backslash\text{qty}\langle\backslash\text{typical}\rangle_{\square} \rightarrow \langle\blacksquare\rangle$ $\backslash\text{qty}=\backslash\text{typical}=\square \rightarrow \blacksquare\ $ $\backslash\text{qty}\big\{\}_{\square} \rightarrow \{\}$ $\backslash\text{qty}\Big\{\}_{\square} \rightarrow \{\}$ $\backslash\text{qty}\bigg\{\}_{\square} \rightarrow \{\}$ $\backslash\text{qty}\Bigg\{\}_{\square} \rightarrow \{\}$ $\backslash\text{pqty}\{\} \leftrightarrow \backslash\text{qty}()$ $\backslash\text{bqty}\{\} \leftrightarrow \backslash\text{qty}[]$ $\backslash\text{vqty}\{\} \leftrightarrow \backslash\text{qty} $ $\backslash\text{Bqty}\{\} \leftrightarrow \backslash\text{qty}\{\}$	automatic () braces automatic [] braces automatic braces automatic { } braces automatic < > braces automatic braces manual sizing (works with any of the above bracket types)
<code>\absolutevalue</code>	$\backslash\text{abs}\{a\} \rightarrow a $ $\backslash\text{abs}\Big\{a\} \rightarrow a $ $\backslash\text{abs}*\{\backslash\text{grande}\} \rightarrow \blacksquare $	automatic sizing; equivalent to <code>\qty a </code> inherits manual sizing syntax from <code>\qty</code> star for no resize
<code>\norm</code>	$\backslash\text{norm}\{a\} \rightarrow \ a\ $ $\backslash\text{norm}\Big\{a\} \rightarrow \ a\ $ $\backslash\text{norm}*\{\backslash\text{grande}\} \rightarrow \ \blacksquare\ $	automatic sizing manual sizing star for no resize
<code>\evaluated</code>	$\backslash\text{eval}\{x\}_0^{\infty} \rightarrow x _0^{\infty}$ $\backslash\text{eval}\Big\{x\}_0^{\infty} \rightarrow x _0^{\infty}$ $\backslash\text{eval}(x _0^{\infty} \rightarrow (x _0^{\infty}$ $\backslash\text{eval}[x _0^{\infty} \rightarrow [x _0^{\infty}$ $\backslash\text{eval}[\backslash\text{venti} _0^{\infty} \rightarrow \left[\blacksquare\right]_0^{\infty}$ $\backslash\text{eval}*\backslash\text{venti} _0^{\infty} \rightarrow [\blacksquare]_0^{\infty}$	vertical bar for evaluation limits alternate form alternate form automatic sizing star for no resize
<code>\order</code>	$\backslash\text{order}\{x^2\} \rightarrow \mathcal{O}(x^2)$ $\backslash\text{order}\Big\{x^2\} \rightarrow \mathcal{O}(x^2)$ $\backslash\text{Order}\{x^2\} \rightarrow \mathcal{O}(x^2)$ $\backslash\text{oorder}\{x^2\} \rightarrow o(x^2)$	order symbol; automatic sizing and space handling manual sizing manual sizing lowercase o

	$\mathcal{O}(x^2) \rightarrow \mathcal{O}(x^2)$	uppercase O
	$\mathcal{O}(\text{grande}) \rightarrow \mathcal{O}(\text{grande})$	star for no resize
$\backslash\text{commutator}$	$\backslash\text{comm}\{A\}\{B\} \rightarrow [A, B]$	automatic sizing
	$\backslash\text{comm}\Big\{A\}\{B\} \rightarrow [A, B]$	manual sizing
	$\backslash\text{comm}\ast\{A\}\{\text{grande}\} \rightarrow [A, \text{grande}]$	star for no resize
$\backslash\text{anticommutator}$	$\backslash\text{acomm}\{A\}\{B\} \rightarrow \{A, B\}$	same as $\backslash\text{poissonbracket}$
$\backslash\text{poissonbracket}$	$\backslash\text{pb}\{A\}\{B\} \rightarrow \{A, B\}$	same as $\backslash\text{anticommutator}$

出于兼容性考量, 仍然保留了 $\backslash\text{matrixquantity}$ ($\backslash\text{mqty}$)、 $\backslash\text{smallmatrixquantity}$ ($\backslash\text{smqty}$) 命令的定义, 它们与 $\backslash\text{quantity}$ 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: $\backslash\text{genegralmatrix}$ 、 $\backslash\text{commamatrix}$ 、 $\backslash\text{diagonalmatrix}$, 及 $\backslash\text{qxmatrix}$ 。

例 1: $\backslash\text{diagonalmatrix}$

```

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

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

```

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

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

例 2: \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 3: \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

```

\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 4:

```
\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 5:

```
\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 6:

```
\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 7:

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

$\commamatrix=[b, align=\T, cr=\N]{a \T b \T c \N d \T e \T f}$ \
{\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` `[[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 8:

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

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

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

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

例 9:

```

\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}})
} }
$\twotwomat[v]{plusmain}$

```

$$\begin{pmatrix} \frac{\partial A_1}{\partial x_1} & \frac{\partial A_1}{\partial x_2} & \cdots & \frac{\partial A_1}{\partial x_n} \\ \frac{\partial A_2}{\partial x_1} & \frac{\partial A_2}{\partial x_2} & \cdots & \frac{\partial A_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial A_n}{\partial x_1} & \frac{\partial A_n}{\partial x_2} & \cdots & \frac{\partial A_n}{\partial x_n} \end{pmatrix} \begin{matrix} A_1(x_1) & A_1(x_2) & \cdots & A_1(x_n) \\ A_2(x_1) & A_2(x_2) & \cdots & A_2(x_n) \\ \vdots & \vdots & \ddots & \vdots \\ A_n(x_1) & A_n(x_2) & \cdots & A_n(x_n) \end{matrix}$$

$$\begin{vmatrix} 1 + A_1(x_1) & A_1(x_2) & \cdots & A_1(x_n) \\ A_2(x_1) & 1 + A_2(x_2) & \cdots & A_2(x_n) \\ \vdots & \vdots & \ddots & \vdots \\ A_n(x_1) & A_n(x_2) & \cdots & 1 + A_n(x_n) \end{vmatrix}$$

一个稍复杂的例子:

例 10:

```

% \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]}
$\hessian[v,sepdim=1ex]{f}{x}{m}$
$\hessian[b,sepdim=1ex]{\mathcal{H}}{p}{n}$

```

$$\begin{vmatrix} \frac{\partial^2 f}{\partial x_1^2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_m \partial x_1} & \cdots & \frac{\partial^2 f}{\partial x_m^2} \end{vmatrix} \begin{vmatrix} \frac{\partial^2 \mathcal{H}}{\partial p_1^2} & \cdots & \frac{\partial^2 \mathcal{H}}{\partial p_1 \partial p_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 \mathcal{H}}{\partial p_n \partial p_1} & \cdots & \frac{\partial^2 \mathcal{H}}{\partial p_n^2} \end{vmatrix}$$

`\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   \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。

4.5 matrix 类通用键值选项

<hr/> array <hr/>		
<hr/> expand <hr/>	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。	
<hr/> rows <hr/>	rows = {整数}	
<hr/> cols <hr/>	cols = {整数}	
<hr/> auto-update <hr/>	auto-update = true false	初始值 = false
	是否自动更新行数和列数。	
<hr/> main <hr/>		
<hr/> row-list <hr/>	设置相应行、列的角标。	
<hr/> col-list <hr/>		
<hr/> infinite <hr/>	是否无穷维矩阵。	
<hr/> !infinite <hr/>		
<hr/> element-code <hr/>	element-code = {<code>}	
<hr/> element-code* <hr/>	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。	
<hr/> element-except <hr/>		
<hr/> element-except+ <hr/>		

<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 = $\langle m p b B v V sm ... \rangle$ 设置输出矩阵的类型。m 使用 <code>matrix</code> 环境, p 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。	初始值 = <code>m</code>
<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> 为真, 则将矩阵元素转置, 不对角标转置。	初始值 = <code>false</code>
<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 是否启用 <code>enhanced</code> 模式。	初始值 = <code>false</code>
<hr/> cr <hr/> align <hr/> sep <hr/>	cr = $\langle 符号 \rangle$ align = $\langle 符号 \rangle$ sep = $\langle 符号 \rangle$ 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 ($\backslash\backslash$) 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。	初始值 = <code>;</code> 初始值 = <code>,</code> 初始值 = <code>,</code>
<hr/> adi-order <hr/>	adi-order = $\langle adi dia iad aid ida dai \rangle$ 设置 <code>array/main</code> 、 <code>diag</code> 、 <code>item</code> 的处理顺序。	初始值 = <code>adi</code>
<hr/> beginning <hr/> beginning+ ending <hr/> 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 11:

```

 $\backslash\commamatrix[
  begin=\begin{array}, end=\end{array}, args*={{\clr}}, \% two {{ }}
]{A, E, I; MNOP, QRST, UVWX}$
 $\backslash\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 12:

```

 $\backslash$ 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 13:

```

 $\backslash$ 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}
}$

```

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

例 14:

```

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

```

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

例 15:

```

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

```

$$\begin{vmatrix} 1 & A & A & A & -1 \\ A & 2 & A & -2 & A \\ A & A & -3 & A & A \\ A & -4 & A & 4 & A \\ -5 & A & A & A & 5 \end{vmatrix} \begin{vmatrix} 1 & \square & A & \square & -1 \\ A & 2 & A & -2 & A \\ A & \square & -3 & \square & A \\ A & -4 & A & 4 & A \\ -5 & A & A & A & 5 \end{vmatrix} \begin{vmatrix} 1 & \square & A & \square & \blacksquare \\ A & 2 & A & -2 & \blacksquare \\ A & \square & -3 & \square & \blacksquare \\ \heartsuit & \heartsuit & \heartsuit & \heartsuit & A \\ -5 & A & A & A & \blacksquare \end{vmatrix}$$

例 16:

```
% 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 `matrix` 等。

`\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`

```

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

```

$\langle data type \rangle$ 为 item 或 diag, 将 $\langle data \rangle$ 保存, 以后可以在 item-data、diag-data 中通过 $\langle data name \rangle$ 来引用。

5.1 定义新的括号命令

```

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

```

由 $\backslash\@declarequantitycmd$ 命令定义的 quantity 命令, 其后除可使用 $\backslash big$ 、 $\backslash Big$ 等命令外, 也支持任意命令。

由 $\backslash\@declareparencmd$ 命令定义的 quantity 命令, 其后仅可使用 $\backslash big$ 、 $\backslash Big$ 、 $\backslash bigg$ 、 $\backslash Bigg$ 这四个命令。

几个例子:

例 17:

```

% \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 18:

```
% \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

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

例 19:

```
\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}$
```

1				1	0	0	0	1		1
2			2	0	2	0	0		2	3
3			3	0	0	3	0		4	5
4			4	0	0	0	4		4	5

1	a			a
2	b			b
c	3			c
d	4			d

\newgeneralmatrix、\NewGeneralMatrix 的例子见例 15 和例 16。