

Introduction to L^AT_EX

Lecture III: Maths

Liu Yihao

SJTU-UMJI Technology Department

June 22, 2021

Use Maths in L^AT_EX

Math Expressions

Math Environments

Spacing in Math Mode

Basic Math Commands

Matrices and Arrays

Useful Maths Packages

Common Packages

The `physics` Package

The `systeme` Package

The `gauss` Package

- 1 Use Maths in L^AT_EX
 - Math Expressions
 - Math Environments
 - Spacing in Math Mode
 - Basic Math Commands
 - Matrices and Arrays

2 Useful Maths Packages

Use Maths in L^AT_EX

Math Expressions

Math Environments

Spacing in Math Mode

Basic Math Commands

Matrices and Arrays

Useful Maths Packages

Common Packages

The `physics` PackageThe `systeme` PackageThe `gauss` Package

Basic equations in L^AT_EX can be easily “programmed”, for example: ¹

Example

```

1 The well known Pythagorean theorem \((x^2 + y^2 = z^2\) was
2 proved to be invalid for other exponents.
3 Meaning the next equation has no integer solutions:
4
5 \[ x^n + y^n = z^n \]
```

The well known Pythagorean theorem $x^2 + y^2 = z^2$ was proved to be invalid for other exponents. Meaning the next equation has no integer solutions:

$$x^n + y^n = z^n$$

¹Some of this part is ported from the tutorial of Overleaf:

Subscripts and Superscripts

The use of superscripts and subscripts is very common in mathematical expressions involving exponents, indexes, and in some special operators.¹

Example

```
1 \[ a_1^2 + a_2^2 = a_3^2 \]
```

$$a_1^2 + a_2^2 = a_3^2$$

Note that here we use `\[` and `\]` to typeset a mathematical expression. You may see many people (including myself in the past) using a pair of `$$` instead. It is a plain-T_EX command, and is nowadays heavily deprecated. See this discussion [▶ Link](#) on Stack Exchange for more information.

¹Some of this part is ported from the tutorial of Overleaf:

If the expression contains long superscripts or subscripts, these need to be collected in braces, as L^AT_EX normally applies the mathematical commands `^` and `_` only to the following character:

Example

```
1 \[ x^{2 \alpha} - 1 = y_{ij} + y_{ij} \]  
2 \[ (a^n)^{r+s} = a^{nr+ns} \]  
3 \[ x^{abc}, \quad \text{\quad} x_{abc}, \quad \text{\quad} x^{abc}_{abc} \]  
4 \[ x^{abc}, \quad \text{\quad} x_{abc}, \quad \text{\quad} x^{abc}_{abc} \]
```

$$x^{2\alpha} - 1 = y_{ij} + y_{ij}$$

$$(a^n)^{r+s} = a^{nr+ns}$$

$$x^{abc}, \quad x_{abc}, \quad x^{abc}_{abc}$$

$$x^{abc}, \quad x_{abc}, \quad x^{abc}_{abc}$$

Brackets and Parentheses

Parentheses and brackets are very common in mathematical formulas. You can easily control the size and style of brackets in \LaTeX .¹

Here's how to type some common math braces and parentheses in \LaTeX :

Type	\LaTeX	Code
Parentheses; round brackets	$(x + y)$	<code>(x+y)</code>
Brackets; square brackets	$[x + y]$	<code>[x+y]</code>
Braces; curly brackets	$\{x + y\}$	<code>\{x+y\}</code>
Angle brackets	$\langle x + y \rangle$	<code>\langle x+y \rangle</code> <code>\rangle</code>
Pipes; vertical bars	$ x + y $	<code>x+y</code> <code>---</code>
Double pipes	$\ x + y\ $	<code>\ x+y\ </code>
Floor brackets	$\lfloor x + y \rfloor$	<code>\lfloor x+y \rfloor</code> <code>\rfloor</code>
Ceil brackets	$\lceil x + y \rceil$	<code>\lceil x+y \rceil</code> <code>\rceil</code>

¹Some of this part is ported from the tutorial of Overleaf: [▶ Link](#)

The size of brackets and parentheses can be manually set, or they can be resized dynamically in your document, as shown in the next example:

Example

```
1 \[ F = G \left( \frac{m_1 m_2}{r^2} \right) \]
```

$$F = G \left(\frac{m_1 m_2}{r^2} \right)$$

Notice that to insert the parentheses or brackets, the `\left` and `\right` commands are used. Even if you are using only one bracket, both commands are mandatory, you can use invisible brackets `\left.` or `\right.` for this.

Example

```
1 \[ \int_a^b x^2 {\rm d} x = \left. \frac{1}{3} x^3 \right|_a^b \]
```

$$\int_a^b x^2 dx = \left. \frac{1}{3} x^3 \right|_a^b$$

Use Maths in L^AT_EX

Math Expressions

Math Environments

Spacing in Math Mode

Basic Math Commands

Matrices and Arrays

Useful Maths Packages

Common Packages

The `physics` PackageThe `systeme` PackageThe `gauss` Package

Sometimes you may want to control the sizes of the brackets yourselves, which is called manually sized brackets. The commands listed are designed for thus purpose.

Size	L ^A T _E X	Code
big	$()$	<code>\big (\big)</code>
Big	$\bigg $	<code>\Big [\Big]</code>
bigg	$\Bigg\}$	<code>\bigg \{ \bigg \}</code>
Bigg	$\Bigg $	<code>\Bigg </code>

Mathematical Modes

L^AT_EX allows two writing modes for mathematical expressions: the `inline` mode and the `display` mode. The first one is used to write formulas that are part of a text. The second one is used to write expressions that are not part of a text or paragraph, and are therefore put on separate lines.

To put your equations in `inline` mode use `\(` and `\)`, `$` and `$` or `\begin{math}` and `\end{math}`. They all work and the choice is a matter of taste.

Example

- 1 In physics, the mass-energy equivalence is stated
- 2 by the equation $E=mc^2$, discovered in 1905 by Albert Einstein.

In physics, the mass-energy equivalence is stated by the equation $E = mc^2$, discovered in 1905 by Albert Einstein.

The `display` mode is usually used with mathematical environments together, which will be discussed in the next subsection.

Numbering of Equations

The `display` mode has two versions: `numbered` and `unnumbered`.

Example

```

1 The mass-energy equivalence is described by the famous equation
2 \[E=mc^2\]
3 discovered in 1905 by Albert Einstein.
4 In natural units ( $c = 1$ ), the formula expresses the identity
5 \begin{equation}
6 E=m
7 \end{equation}

```

The mass-energy equivalence is described by the famous equation

$$E = mc^2$$

discovered in 1905 by Albert Einstein. In natural units ($c = 1$), the formula expresses the identity

$$E = m \tag{1}$$

Use Maths in L^AT_EX

Math Expressions

Math Environments

Spacing in Math Mode

Basic Math Commands

Matrices and Arrays

Useful Maths Packages

Common Packages

The `physics` Package

The `systeme` Package

The `gauss` Package

- 1 Use Maths in L^AT_EX
 - Math Expressions
 - **Math Environments**
 - Spacing in Math Mode
 - Basic Math Commands
 - Matrices and Arrays

2 Useful Maths Packages

The `equation` Environment

An `equation` environment contains a set of maths equations

Command

```
1 \begin{equation(*)}
2 % ...
3 \end{equation(*)}
```

Example

$$\operatorname{rot} F = \left(\frac{\partial F_z}{\partial y} - \frac{\partial F_y}{\partial z} \right) \hat{n}_x + \left(\frac{\partial F_x}{\partial z} - \frac{\partial F_z}{\partial x} \right) \hat{n}_y + \left(\frac{\partial F_y}{\partial x} - \frac{\partial F_x}{\partial y} \right) \hat{n}_z \quad (2)$$

If a star(*) is added, the sequence number of the equation won't be displayed (this feature is from the `amsmath` package, and should behave very similar as directly using `\[` and `\]`). Note that the environment name in the `\begin` and `\end` statements must be the same (both or neither have a * here).

Use Maths in L^AT_EX

Math Expressions

Math Environments

Spacing in Math Mode

Basic Math Commands

Matrices and Arrays

Useful Maths Packages

Common Packages

The `physics` PackageThe `systeme` PackageThe `gauss` Package

The L^AT_EX script of the equation above is quite long, but not so difficult as you think so. All of the useless spaces are omitted, so please pay attention to the necessary spaces (marked in `\`).

```

1 \begin{equation}
2   \mathop{\rm rot}F=\left(\frac{\partial F_z}{\partial y}
3     -\frac{\partial F_y}{\partial z}\right)\hat{n}_x
4     +\left(\frac{\partial F_x}{\partial z}
5     -\frac{\partial F_z}{\partial x}\right)\hat{n}_y
6     +\left(\frac{\partial F_y}{\partial x}
7     -\frac{\partial F_x}{\partial y}\right)\hat{n}_z
8 \end{equation}
```

In math environments, unlike in plain text, normal spaces will not lead to visible spaces in output. Only `\` or `\quad`, `\qquad` etc. will create spaces between words.

`\partial` prints the symbol ∂ , `\frac{...}{...}` makes a fraction.

`\left(` and `\right(` make braces that fit the equation's height.

It is written in plain-L^AT_EX, and things can even be easier with packages like `physics`, which will be demonstrated later.

The `split` Environment (inline)

In order to deal with extremely long equations or equation with multiple lines, we can use the `split` environment. It is an `inline` environment being used in other maths environments.

Example

```

1 \begin{equation}
2   \begin{split}
3     F &= 1+2+3+4+5 \\
4     &= 15
5   \end{split}
6 \end{equation}
```

$$F = 1 + 2 + 3 + 4 + 5 = 15 \quad (3)$$

`&` is used to align the equal marks, and `\\` is used to split the equation into two lines. Only one equation number will be generated in an `equation` environment.

The `split` environment is designed to serve as the entire body of an equation, or an entire line of an `align` or `gather` environment. There cannot be any printed material before or after it within the same enclosing structure.

The `aligned` Environment (inline)

For linear equation systems, the `aligned` environment can be used, which is similar to the `split` environment above. It is also an `inline` environment, which can be used in `inline` mode such as `$$`! Here `split` doesn't work because `\left` and `\right` is an enclosing structure. See this discussion [▶ Link](#) for more information.

Example

```

1 Equations:
2 $
3 \left\lbrace\begin{aligned}
4   x+y &= 1 \ \&\ x-y &= 1
5 \end{aligned}\right.
6 \Longrightarrow
7 \left\lbrace\begin{aligned}
8   x &= 1 \ \&\ y &= 0
9 \end{aligned}\right.
10 $

```

$$\text{Equations: } \begin{cases} x + y = 1 \\ x - y = 1 \end{cases} \implies \begin{cases} x = 1 \\ y = 0 \end{cases}$$

Actually things can also be easier with packages like `systeme`, which will be demonstrated later.

The `align` Environment

An `align` environment can be used to simply the `split` or `aligned` in the `equation` environment. But it numbers the equation on each line.

Example

```
1 \begin{align}
2   F &= 1+2+3+4+5 \\
3   &= 15 \\
4 \end{align}
```

$$F = 1 + 2 + 3 + 4 + 5 \quad (4)$$

$$= 15 \quad (5)$$

Use `align*` so that there will be no number(s).

Example

```
1 \begin{align*}
2   a+b &\Leftrightarrow b+a \\
3   (a+b)+c &\Leftrightarrow a+(b+c) \\
4 \end{align*}
```

$$a + b \Leftrightarrow b + a$$

$$(a + b) + c \Leftrightarrow a + (b + c)$$

Use Maths in L^AT_EX

Math Expressions

Math Environments

Spacing in Math Mode

Basic Math Commands

Matrices and Arrays

Useful Maths Packages

Common Packages

The `physics` Package

The `systeme` Package

The `gauss` Package

The ampersand character `&` determines where the equations align. The odd columns are right-aligned, and the even ones are left-aligned, so you can use `&&` if you want to make two neighbor column aligned to the same direction.

Example

```
1 \begin{align*}
2 \text{\texttt{\texttt{(right)}}} & \text{\texttt{\texttt{(left)}}} && \text{\texttt{\texttt{(left)}}} & \text{\texttt{\texttt{(right)}}} \\
3 && \text{\texttt{\texttt{(right)}}} & \text{\texttt{\texttt{(left)}}} & \\
4 x & & =y & & w & & =z & & a&=b+c \\
5 2x & & =-y & & 3w & & =z/2 & & a&=b \\
6 -4+5x & & =2+y & & w+2 & & =-1+w & & ab&=cb \\
7 \end{align*}
```

(right)(left)	(left)	(right)	(right)(left)
$x = y$	w	$= z$	$a = b + c$
$2x = -y$	$3w$	$= z/2$	$a = b$
$-4 + 5x = 2 + y$	$w + 2$	$= -1 + w$	$ab = cb$

The `cases` Environment (inline)

The linear system of equations can also be typeset simply with the `cases` environment. It is less flexible than an `aligned` environment, eg., there can only be one `&` on each row. Another minor difference is that the horizontal space before `&` is larger than other similar environments.

Example

```

1  \begin{equation}
2  \left\{\begin{aligned}
3  x+y &= 1 \\
4  x-y &= 1
5  \end{aligned}\right.
6  \end{equation}
7
8  \begin{equation}
9  \begin{cases}
10 x+y &= 1 \\
11 x-y &= 1
12 \end{cases}
13 \end{equation}
```

$$\begin{cases} x + y = 1 \\ x - y = 1 \end{cases} \quad (6)$$

$$\begin{cases} x + y = 1 \\ x - y = 1 \end{cases} \quad (7)$$

The `gather` Environment

Use Maths in L^AT_EX

Math Expressions

Math Environments

Spacing in Math Mode

Basic Math Commands

Matrices and Arrays

Useful Maths Packages

Common Packages

The `physics` PackageThe `systeme` PackageThe `gauss` Package

If you just need to display a set of consecutive equations, centered and with no alignment whatsoever, use the `gather` environment. The asterisk trick to set/unset the numbering of equations also works here.

Example

```
1 \begin{gather}
2   2x - 5y = 8 \\
3   3x^2 + 9y = 3a + c
4 \end{gather}
```

$$2x - 5y = 8 \quad (8)$$

$$3x^2 + 9y = 3a + c \quad (9)$$

Use Maths in L^AT_EX

Math Expressions

Math Environments

Spacing in Math Mode

Basic Math Commands

Matrices and Arrays

Useful Maths Packages

Common Packages

The `physics` PackageThe `systeme` PackageThe `gauss` Package

The `gathered` Environment (inline)

There is also an `inline` version of `gather`, called `gathered`. The relationship of them is similar to `align` and `aligned`.

Example

```
1 \begin{equation}
2   \begin{gathered}
3     2x - 5y = 8 \\
4     3x^2 + 9y = 3a + c
5   \end{gathered}
6 \end{equation}
```

$$\begin{aligned} 2x - 5y &= 8 \\ 3x^2 + 9y &= 3a + c \end{aligned} \tag{10}$$

The `multline` Environment

For equations longer than a line use the `multline` environment. Insert a double backslash to set a point for the equation to be broken. The first part will be aligned to the left and the second part will be displayed in the next line and aligned to the right.

Example

```

1 \begin{multline}
2   p(x) = 3x^6 + 14x^5y + 590x^4y^2 + 19x^3y^3 \\
3         - 12x^2y^4 - 12xy^5 + 2y^6 - a^3b^3
4 \end{multline}

```

$$\begin{aligned}
 p(x) = & 3x^6 + 14x^5y + 590x^4y^2 + 19x^3y^3 \\
 & - 12x^2y^4 - 12xy^5 + 2y^6 - a^3b^3 \quad (11)
 \end{aligned}$$

The equation number will be in the last line, use `multline*` for no numbering.

Use Maths in L^AT_EX

Math Expressions

Math Environments

Spacing in Math Mode

Basic Math Commands

Matrices and Arrays

Useful Maths Packages

Common Packages

The `physics` PackageThe `systeme` PackageThe `gauss` Package

For equations equal or longer than three lines,

Example

```
1 \begin{multline*}
2   a+b+c=1 \\
3   b+c=2 \\
4   c+d=1 \\
5   d=3
6 \end{multline*}
```

$$a + b + c = 1$$

$$b + c = 2$$

$$c + d = 1$$

$$d = 3$$

Here, the first column is left-aligned, the last column is right-aligned and the others ones are center-aligned.

The `flalign` Environment

For equations aligned left, use the `flalign` environment. It is similar to the `align` environment.

Example

```

1 \begin{flalign}
2   a+b &=1=& b+a \\
3   b   &=2=& c
4 \end{flalign}

```

$$a + b = 1 =$$

$$b + a \quad (12)$$

$$b = 2 =$$

$$c \quad (13)$$

You may notice that the columns are flushed left (start from the left most position) and the right most column is flushed right, different from the `align` environment.

Use Maths in L^AT_EX

Math Expressions

Math Environments

Spacing in Math Mode

Basic Math Commands

Matrices and Arrays

Useful Maths Packages

Common Packages

The `physics` Package

The `systeme` Package

The `gauss` Package

- 1 Use Maths in L^AT_EX
 - Math Expressions
 - Math Environments
 - Spacing in Math Mode
 - Basic Math Commands
 - Matrices and Arrays

2 Useful Maths Packages

Horizontal spacing in maths mode is useful in several situations, let's see an example: ¹

Example

```

1 Assume we have the next sets
2 \[
3 S = \{ z \in \mathbb{C} \setminus , | \setminus , |z| < 1 \setminus \} \quad \text{\texttt{\textcolor{teal}{S}}}
4 \text{\texttt{\textcolor{red}{and}}} \quad S_2=\partial\{S\}
5 \]

```

Assume we have the next sets

$$S = \{z \in \mathbb{C} \mid |z| < 1\} \quad \text{and} \quad S_2 = \partial S$$

As you see in this example, a mathematical text can be explicitly spaced by means of some special commands.

¹Some of this part is ported from the tutorial of Overleaf:

Use Maths in L^AT_EX

Math Expressions

Math Environments

Spacing in Math Mode

Basic Math Commands

Matrices and Arrays

Useful Maths Packages

Common Packages

The `physics` PackageThe `systeme` PackageThe `gauss` Package

The spacing depends on the command you insert, the example below contains a complete list of spaces and how they look like.

Example

```

1 \begin{align*}
2 f(x) =& x^2\! + 3x\! + 2 \quad \! \\
3 f(x) =& x^2+3x+2 \quad \! \\
4 f(x) =& x^2\,, +3x\,, +2 \quad \! \\
5 f(x) =& x^2\!: +3x\!: +2 \quad \! \\
6 f(x) =& x^2\!; +3x\!; +2 \quad \! \\
7 f(x) =& x^2\! +3x\! +2 \quad \! \\
8 f(x) =& x^2\!\quad +3x\!\quad +2 \quad \! \\
9 f(x) =& x^2\!\quad\quad +3x\!\quad\quad +2 \\
10 \end{align*}

```

$$f(x) = x^2 + 3x + 2$$

$$f(x) = x^2 + 3x + 2$$

$$f(x) = x^2 + 3x + 2$$

$$f(x) = x^2 + 3x + 2$$

$$f(x) = x^2 + 3x + 2$$

$$f(x) = x^2 + 3x + 2$$

$$f(x) = x^2 + 3x + 2$$

$$f(x) = x^2 + 3x + 2$$

Vertical Spacing

When the space between `display` maths and the main body paragraph is considered larger than expectation, is there any way to modify the line spacing?

In default style of `display` mode is like

Example

```

1 your body paragraph is supposed to be typed here
2 \begin{equation}
3   a \times b = c
4 \end{equation}
5 your body paragraph is supposed to be typed here

```

your body paragraph is supposed to be typed here

$$a \times b = c \tag{14}$$

your body paragraph is supposed to be typed here

Use Maths in L^AT_EX

Math Expressions

Math Environments

Spacing in Math Mode

Basic Math Commands

Matrices and Arrays

Useful Maths Packages

Common Packages

The `physics` PackageThe `systeme` PackageThe `gauss` Package

You can use `\setlength` to set the `displayskip`.

Command

```
1 \setlength\abovedisplayskip{<length>}
2 \setlength\belowdisplayskip{<length>}
```

Example

```
1 \setlength\abovedisplayskip{0em}
2 \setlength\belowdisplayskip{0em}
3 your body paragraph is supposed to be typed here
4 \begin{equation}
5   a \times b = c
6 \end{equation}
7 your body paragraph is supposed to be typed here
```

your body paragraph is supposed to be typed here

$$a \times b = c$$

(15)

your body paragraph is supposed to be typed here

Use Maths in L^AT_EX

Math Expressions

Math Environments

Spacing in Math Mode

Basic Math Commands

Matrices and Arrays

Useful Maths Packages

Common Packages

The `physics` Package

The `systeme` Package

The `gauss` Package

- 1 Use Maths in L^AT_EX
 - Math Expressions
 - Math Environments
 - Spacing in Math Mode
 - **Basic Math Commands**
 - Matrices and Arrays

2 Useful Maths Packages

Use Maths in L^AT_EX

Math Expressions

Math Environments

Spacing in Math Mode

Basic Math Commands

Matrices and Arrays

Useful Maths Packages

Common Packages

The `physics` PackageThe `systeme` PackageThe `gauss` Package

Fractions and Binomials

Fractions and binomial coefficients are common mathematical elements with similar characteristics - one number goes on top of another. ¹

Command

```
1 \frac{top}{bottom} % fraction
2 \binom{top}{bottom} % binomial coefficients
```

Using fractions and binomial coefficients in an expression is straightforward.

Example

```
1 The binomial coefficient is defined by the next expression:
2 \[ \binom{n}{k} = \frac{n!}{k!(n-k)!} \]
```

The binomial coefficient is defined by the next expression:

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}$$

¹Some of this part is ported from the tutorial of Overleaf:

Use Maths in L^AT_EX

Math Expressions

Math Environments

Spacing in Math Mode

Basic Math Commands

Matrices and Arrays

Useful Maths Packages

Common Packages

The `physics` PackageThe `systeme` PackageThe `gauss` Package

In `inline` and `display` mode, the appearance of the fractions and binomials may differ. You can use `\displaystyle` or `\textstyle` to adjust the size of the fractions and binomials, or use `\dfrac` if not all fractions in an equation need to be resized.

Example

```

1 When displaying fractions in-line, for example  $\frac{3x}{2}$ 
2 you can set a different display style:  $ \displaystyle \frac{3x}{2} $ .
3 Or you can use  $\dfrac{3x}{2}$ . This is also true the other way around
4  \[ f(x)=\binom{n}{x}=\frac{n!}{x!(n-x)!} \quad \text{and} \quad 
5  f(x)=\textstyle\binom{n}{x}=\frac{n!}{x!(n-x)!} \]

```

When displaying fractions in-line, for example $\frac{3x}{2}$ you can set a different display style: $\frac{3x}{2}$. Or you can use $\frac{3x}{2}$. This is also true the other way around

$$f(x) = \binom{n}{x} = \frac{n!}{x!(n-x)!} \quad \text{and} \quad f(x) = \binom{n}{x} = \frac{n!}{x!(n-x)!}$$

The command `\displaystyle` will format the fractions and binomials as if they were in mathematical display mode. On the other side, `\textstyle` will change the style of them as if they were part of the text.

Use Maths in L^AT_EX

Math Expressions

Math Environments

Spacing in Math Mode

Basic Math Commands

Matrices and Arrays

Useful Maths Packages

Common Packages

The `physics` PackageThe `systeme` PackageThe `gauss` Package

The usage of fractions is quite flexible, they can be nested to obtain more complex expressions. And `\cfrac` can be used to make continued fractions.

Example

- 1 The fractions can be nested
- 2 `\[\frac{1+\frac{a}{b}}{1+\frac{1}{1+\frac{1}{a}}} \]`
- 3 Now a wild example
- 4 `\[a_0+\cfrac{1}{a_1+\cfrac{1}{a_2+\cfrac{1}{a_3+\cdots}}} \]`

The fractions can be nested

$$\frac{1 + \frac{a}{b}}{1 + \frac{1}{1 + \frac{1}{a}}}$$

Now a wild example

$$a_0 + \frac{1}{a_1 + \frac{1}{a_2 + \frac{1}{a_3 + \cdots}}}$$

Operators

Characters in mathematical mode are usually shown in italics, but sometimes especial function names require different formatting (font and skip), this is accomplished by using operators defined in L^AT_EX.¹

Trigonometrical functions, logarithms, and some others can be written in a document by means of some special commands.

Example

```
1 \[ \sin(a + b) = \sin(a)\cos(b) + \cos(a)\sin(b) \]
2 \[ \log_a b = \frac{\log_c b}{\log_c a} = \frac{\ln b}{\ln a} \]
3 \[ \tan a, \quad \arccos a, \quad \arcsin a, \quad \arctan a \]
```

$$\sin(a + b) = \sin(a) \cos(b) + \cos(a) \sin(b)$$

$$\log_a b = \frac{\log_c b}{\log_c a} = \frac{\ln b}{\ln a}$$

$$\tan a, \quad \arccos a, \quad \arcsin a, \quad \arctan a$$

¹Some of this part is ported from the tutorial of Overleaf:

Use Maths in L^AT_EX

Math Expressions

Math Environments

Spacing in Math Mode

Basic Math Commands

Matrices and Arrays

Useful Maths Packages

Common Packages

The `physics` PackageThe `systeme` PackageThe `gauss` Package

Integrals

Integral expression can be added using the command

Command

```
1 \int_{lower}^{upper}
```

Note, that integral expression may seems a little different in `inline` and `display` math mode - in `inline` mode the integral symbol and the limits are compressed.

Example

```
1 Integral $\int_{a}^{b} x^2 dx$
   ↳ inside text
2 \[ \int_{a}^{b} x^2 dx \]
```

Integral $\int_a^b x^2 dx$ inside text

$$\int_a^b x^2 dx$$

There is always an argue about whether *italic* or roman style of “d” should be used in integrals and derivatives. There’s no right or wrong. If you prefer to use roman style, try `commath` or `physics` package. Either of them provides some macros to insert the “d” you want simply.

Use Maths in L^AT_EX

Math Expressions

Math Environments

Spacing in Math Mode

Basic Math Commands

Matrices and Arrays

Useful Maths Packages

Common Packages

The `physics` PackageThe `systeme` PackageThe `gauss` Package

Multiple Integrals

To obtain double/triple/multiple integrals you must use `amsmath` package.

Example

```

1 \begin{gather*}
2 \iint_V \mu(u,v) \, du \, dv \\
3 \iiint_V \mu(u,v,w) \, du \, dv \, dw \\
4 \iiint_V \mu(t,u,v,w) \\
\quad \rightarrow \, dt \, du \, dv \, dw \\
5 \idotsint_V \mu(u_1, \dots, u_k) \, du_1 \\
\quad \rightarrow \, \dots du_k \\
6 \end{gather*}

```

$$\iint_V \mu(u, v) \, du \, dv$$

$$\iiint_V \mu(u, v, w) \, du \, dv \, dw$$

$$\iiint_V \mu(t, u, v, w) \, dt \, du \, dv \, dw$$

$$\int \cdots \int_V \mu(u_1, \dots, u_k) \, du_1 \cdots du_k$$

Cyclic Integrals

To obtain cyclic integrals you must use `esint` package.

Example

```
1 \begin{gather*}
2 \oint_V f(s) \, ds \\
3 \oiint_V f(s,t) \, ds \, dt \\
4 \end{gather*}
```

$$\oint_V f(s) \, ds$$
$$\oiint_V f(s,t) \, ds \, dt$$

Limits, Sums and Products

Like integrals, limits, sums and products expression are compressed in `inline` mode.

Command

```
1 \limits_{lower}
2 \sum_{lower}^{upper}
3 \prod_{lower}^{upper}
```

Example

```
1 Limit $\lim_{x\to\infty} f(x)$ inside text^^I
2 \[ \lim_{x\to\infty} f(x) \]
```

Limit $\lim_{x \rightarrow \infty} f(x)$ inside text

$$\lim_{x \rightarrow \infty} f(x)$$

Example

- 1 Sum `$\sum_{n=1}^{\infty} 2^{-n} = 1$` inside text
- 2 `\[\sum_{n=1}^{\infty} 2^{-n} = 1 \]`

Sum $\sum_{n=1}^{\infty} 2^{-n} = 1$ inside text

$$\sum_{n=1}^{\infty} 2^{-n} = 1$$

Example

- 1 Product `$\prod_{i=a}^b f(i)$` inside text
- 2 `\[\prod_{i=a}^b f(i) \]`

Product $\prod_{i=a}^b f(i)$ inside text

$$\prod_{i=a}^b f(i)$$

Improvement of Integrals, Limits, Sums and Products

In `inline` math mode the integral/sum/product lower and upper limits are placed right of integral symbol. Similar is for limit expressions. If you want the limits of an integral/sum/product to be specified above and below the symbol in `inline` math mode (or in `display` mode), use the `\limits` command before limits specification.

Example

- 1 Integral `\int_{a}^{b} x^2 dx` inside text `\par`
- 2 Improved integral `\int\limits_{a}^{b} x^2 dx` inside text `\par`
- 3 Use limits in display mode `\[\int\limits_{a}^{b} x^2 dx \]`

Integral $\int_a^b x^2 dx$ inside text

Improved integral $\int\limits_a^b x^2 dx$ inside text

Use limits in display mode

$$\int_a^b x^2 dx$$

Use Maths in L^AT_EX

Math Expressions

Math Environments

Spacing in Math Mode

Basic Math Commands

Matrices and Arrays

Useful Maths Packages

Common Packages

The `physics` PackageThe `systeme` PackageThe `gauss` Package

Moreover, adding `\displaystyle` beforehand will make the symbol in inline mode large and easier to read, as in display mode.

Example

- 1 Limit `$\lim_{x\to\infty} f(x)$` inside text `\par`
- 2 Display style limit `$\displaystyle\lim_{x\to\infty} f(x)$` inside text

Limit $\lim_{x \rightarrow \infty} f(x)$ inside text

Display style limit $\lim_{x \rightarrow \infty} f(x)$ inside text

On the other hand, `\mathlarger` command (provided by `relsize` package) is used to get bigger integral symbol in display.

Example

- 1 `$\int \frac{1}{2} dx - \mathlarger{\int \frac{1}{2} dx}$`

$$\int \frac{1}{2} dx - \int \frac{1}{2} dx$$

Other Math Symbols

Some examples of other common used math symbols are shown.

Name	\LaTeX	Code
Square Root	$\sqrt{a} \sqrt[b]{a}$	<code>\sqrt {a}\ \ \sqrt [b]{a}</code>
Over/Under Line	$\overline{a+b} \quad \underline{a+b}$	<code>\overline {a+b}\ \ \underline {a+b}</code>
Over Brace	$\overbrace{1+2+\cdots+n}$	<code>\overbrace {1+2+\cdots +n}^n</code>
Under Brace	$\underbrace{1+2+\cdots+n}$	<code>\underbrace {1+2+\cdots +n}_n</code>
Over Arrow	$\overrightarrow{a+b} \quad \overleftarrow{a+b}$	<code>\overrightarrow {a+b}\ \ \overleftarrow {a+b}</code>
Under Arrow	$\underrightarrow{a+b} \quad \underleftarrow{a+b}$	<code>\underrightarrow {a+b}\ \ \underleftarrow {a+b}</code>
Dots	$\dots \quad \cdots \quad \vdots \quad \ddots$	<code>\dots \ \ \cdot \ \ \cdots \ \ \vdots \ \ \ddots</code>
Arrows	$\rightarrow \quad \leftarrow \quad \leftrightarrow$ $\Rightarrow \quad \Leftarrow \quad \Leftrightarrow$ $\longleftarrow \quad \Longrightarrow$	<code>\rightarrow \ \ \leftarrow \ \ \leftrightarrow</code> <code>\Rightarrow \ \ \Leftarrow \ \ \Leftrightarrow</code> <code>\longleftarrow \ \ \Longrightarrow</code>

Use Maths in L^AT_EX

Math Expressions

Math Environments

Spacing in Math Mode

Basic Math Commands

Matrices and Arrays

Useful Maths Packages

Common Packages

The `physics` PackageThe `systeme` PackageThe `gauss` Package

Mathematical Fonts

In mathematical mode as well as in text mode, you can change the typeface as needed. For instance, it's customary to represent real numbers with a blackboard bold font, or topological spaces with calligraphic font.¹

For some elements is convenient to have the possibility of changing the font typeface.

Example

```

1 Let \(\mathcal{T}\) be a topological space, a basis is defined as
2 \[ \mathcal{B} = \{B_{\alpha} \mid B_{\alpha} \in \mathcal{T}, \bigcup_{\alpha} B_{\alpha} = U, U \in \mathcal{T}\}
3 U = \bigcup_{\alpha} B_{\alpha} \text{ for all } U \in \mathcal{T} \]

```

Let \mathcal{T} be a topological space, a basis is defined as

$$\mathcal{B} = \{B_{\alpha} \in \mathcal{T} \mid U = \bigcup_{\alpha} B_{\alpha} \forall U \in \mathcal{T}\}$$

¹Some of this part is ported from the tutorial of Overleaf:

Mathematical Fonts for Capital Letters

There are some font typefaces that support only a limited number of characters; these fonts usually denote some special sets.

Example

1	<code>\begin{gather*}</code>	
2	<code>RQSZ \\\</code>	<i>RQSZ</i>
3	<code>\mathcal{RQSZ} \\\</code>	<i>RQSZ</i>
4	<code>\mathfrak{RQSZ} \\\</code>	<i>RQSZ</i>
5	<code>\mathbb{RQSZ}</code>	RQSZ
6	<code>\end{gather*}</code>	RQSZ

This example shows Calligraphic, Fraktur and Blackboard bold typefaces. For instance, to display the R in blackboard bold typeface `\mathbb{R}` will do the trick.

Other Mathematical Fonts

It's possible to set a different font family for a complete mathematical expression.

Example

```

1 \begin{gather*}
2   3x^2 \in R \subset Q \\
3   \mathnormal{3x^2 \in R \subset Q} \\
4   \mathrm{3x^2 \in R \subset Q} \\
5   \mathit{3x^2 \in R \subset Q} \\
6   \mathbf{3x^2 \in R \subset Q} \\
7   \mathsf{3x^2 \in R \subset Q} \\
8   \mathtt{3x^2 \in R \subset Q} \\
9 \end{gather*}

```

$$3x^2 \in R \subset Q$$

$$3x^2 \in R \subset Q$$

$$3x^2 \in R \subset Q$$

$$3x^2 \in R \subset Q$$

$$\mathbf{3x^2 \in R \subset Q}$$

$$3x^2 \in R \subset Q$$

$$3x^2 \in R \subset Q$$

In this case, not only letters but all characters change its appearance, for example `\mathit{3x^2}` italicises the entire expression.

Use Maths in L^AT_EX

Math Expressions

Math Environments

Spacing in Math Mode

Basic Math Commands

Matrices and Arrays

Useful Maths Packages

Common Packages

The `physics` PackageThe `systeme` PackageThe `gauss` Package

Define Own Symbols

If you need to add a personalized operator to be displayed in Roman font instead of italics use `\DeclareMathOperator`, provided by the the package `amsmath`.

Example

```
1 \DeclareMathOperator{\Mr}{M_{\mathbb{R}}}
2 User-defined operator for matrices with Real entries $ x \in \Mr $
```

User-defined operator for matrices with Real entries $x \in M_{\mathbb{R}}$

The command can be slightly modified if you need that your defined operator uses subscripts, as the `\lim` operator, in such case use `\DeclareMathOperator*`.

You can also use `\mathop` to define a italics math operator supporting subscripts, and change it to Roman font by hand.

Example

```
1 \[ \mathop{\mathrm{limsup}}_{n \rightarrow \infty} \mathop{\mathrm{rot}} F_n \]
```

$$\limsup_{n \rightarrow \infty} \operatorname{rot} F_n$$

Use Maths in L^AT_EX

Math Expressions

Math Environments

Spacing in Math Mode

Basic Math Commands

Matrices and Arrays

Useful Maths Packages

Common Packages

The **physics** Package

The **systeme** Package

The **gauss** Package

- 1 Use Maths in L^AT_EX
 - Math Expressions
 - Math Environments
 - Spacing in Math Mode
 - Basic Math Commands
 - **Matrices and Arrays**

2 Useful Maths Packages

The `matrix` Environment (inline)

There are various kinds of matrix environments defined in `amsmath` package, they are `matrix`, `pmatrix`, `bmatrix`, `Bmatrix`, `vmatrix`, `Vmatrix`.

Command

```

1 \begin{[p/b/B/v/V]matrix}
2   a_{11} & a_{12} & \dots & a_{1n} \\
3   a_{21} & a_{22} & \dots & a_{2n} \\
4   \dots & \dots & \dots & \dots \\
5   a_{n1} & a_{n2} & \dots & a_{nn} \\
6 \end{[p/b/B/v/V]matrix}

```

Example

```

1 \begin{equation}
2   \begin{pmatrix}
3     a_{11} & a_{12} & a_{13} \\
4     a_{21} & a_{22} & a_{23} \\
5     a_{31} & a_{32} & a_{33}
6   \end{pmatrix}
7 \end{equation}

```

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \quad (16)$$

Use Maths in L^AT_EX

Math Expressions

Math Environments

Spacing in Math Mode

Basic Math Commands

Matrices and Arrays

Useful Maths Packages

Common Packages

The **physics** PackageThe **systeme** PackageThe **gauss** Package

Here is some examples of the style of these matrix.

Example

matrix

$$\begin{matrix} a & b \\ c & d \end{matrix}$$

bmatrix

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

vmatrix

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

pmatrix

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

Bmatrix

$$\begin{Bmatrix} a & b \\ c & d \end{Bmatrix}$$

Vmatrix

$$\begin{Vmatrix} a & b \\ c & d \end{Vmatrix}$$

Some packages may also help simplify the typesetting of matrix, for example, there is some macros defined in the **physics** package to make identity matrix, or generate the examples above more simply.

Use Maths in L^AT_EX

Math Expressions

Math Environments

Spacing in Math Mode

Basic Math Commands

Matrices and Arrays

Useful Maths Packages

Common Packages

The `physics` PackageThe `systeme` PackageThe `gauss` Package

If you need to create matrices with different delimiters, you can add them manually to a plain matrix. For example:

Example

```
1 \begin{equation}
2 \left\lceil
3 \begin{matrix}
4 1 & 2 & 3 \\
5 a & b & c
6 \end{matrix}
7 \right\rceil
8 \end{equation}
9
10 \begin{equation}
11 \left\langle
12 \begin{matrix}
13 1 & 2 & 3 \\
14 a & b & c
15 \end{matrix}
16 \right\rangle
17 \end{equation}
```

$$\left[\begin{matrix} 1 & 2 & 3 \\ a & b & c \end{matrix} \right] \quad (17)$$

$$\left\langle \begin{matrix} 1 & 2 & 3 \\ a & b & c \end{matrix} \right\rangle \quad (18)$$

The `smallmatrix` Environment

When typesetting inline math, the usual `matrix` environments above may look too big. It may be better to use `smallmatrix` in such situations, although you will need to provide your own delimiters.

Example

```

1  Trying to typeset an inline matrix here
2  \begin{pmatrix}
3      a & b \\
4      c & d
5  \end{pmatrix}
6  but it looks too big, so let's try
7  \big(\begin{smallmatrix}
8      a & b \\
9      c & d
10 \end{smallmatrix}\big) instead.
```

Trying to typeset an inline matrix here $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$ but it looks too big, so let's try $\big(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\big)$ instead.

The `array` Environment

An `array` environment is actually a math mode `tabular` environment, and the usage of them are almost the same. You can refer to the lecture about tables for this part.

A simple example is given here:

Example

```

1 \begin{equation}
2   \chi(\lambda) =
3   \left| \begin{array}{ccc}
4     \lambda - a & -b & -c \\
5     -d & \lambda - e & -f \\
6     -g & -h & \lambda - i
7   \end{array} \right|
8 \end{equation}

```

$$\chi(\lambda) = \begin{vmatrix} \lambda - a & -b & -c \\ -d & \lambda - e & -f \\ -g & -h & \lambda - i \end{vmatrix} \quad (19)$$

Use Maths in \LaTeX

Math Expressions

Math Environments

Spacing in Math Mode

Basic Math Commands

Matrices and Arrays

Useful Maths Packages

Common Packages

The `physics` Package

The `systeme` Package

The `gauss` Package

1 Use Maths in \LaTeX

2 Useful Maths Packages

- Common Packages

- The `physics` Package

- The `systeme` Package

- The `gauss` Package

The $\text{AMS-}\text{\LaTeX}$ Packages

AMS- \LaTeX is a collection of \LaTeX document classes and packages developed for the American Mathematical Society (AMS).

It is an extension of plain- \LaTeX maths, with many new maths environments (most of them were introduced in the previous section), maths symbols and maths fonts.

Usually you can insert all of the commands in the preamble of your document.

Command

```
1 \usepackage{amsmath} % loads maths environments
2 \usepackage{amssymb} % loads maths symbols
3 \usepackage{amsfonts} % loads maths fonts
```

Some Other Packages

Recall that we also use some other packages in this lecture:

Command

```
1 \usepackage{esint}    % for cyclic integrals
2 \usepackage{relsize}  % for \mathlarger
```

For a better `array` environment, though it's not mandatory (you can use it without the package), you're recommended to add the `array` package.

Command

```
1 \usepackage{array}
```

Use Maths in L^AT_EX

Math Expressions

Math Environments

Spacing in Math Mode

Basic Math Commands

Matrices and Arrays

Useful Maths Packages

Common Packages

The **physics** Package

The **systeme** Package

The **gauss** Package

1 Use Maths in L^AT_EX

2 Useful Maths Packages

- Common Packages
- The **physics** Package
- The **systeme** Package
- The **gauss** Package

The `physics` Package

To use the `physics` package, simply insert the command in the preamble of your document.

Command

```
1 \usepackage{physics}
```

The goal of this package is to make typesetting equations for `physics` simpler, faster, and more human-readable. But it can also be used in various maths circumstances.

To that end, the commands included in this package have names that make the purpose of each command immediately obvious and remove any ambiguity while reading and editing `physics` code.

The documentation of the `physics` package can be found in <http://mirrors.ctan.org/macros/latex/contrib/physics/physics.pdf>.

Recall the equation:

```

1 \begin{equation}
2   \mathop{\rm rot} F = \left( \frac{\partial F_z}{\partial y} - \frac{\partial F_y}{\partial z} \right) \hat{n}_x
3   + \left( \frac{\partial F_x}{\partial z} - \frac{\partial F_z}{\partial x} \right) \hat{n}_y
4   + \left( \frac{\partial F_y}{\partial x} - \frac{\partial F_x}{\partial y} \right) \hat{n}_z
5   \end{equation}

```

Now we can rewrite it with the commands defined in the `physics` package.

Example

```











1 \begin{equation}
2   \mathop{\rm rot} F = \qty(\pdv{F_z}{y} - \pdv{F_y}{z}) \hat{n}_x +
3   \qty(\pdv{F_x}{z} - \pdv{F_z}{x}) \hat{n}_y +
4   \qty(\pdv{F_y}{x} - \pdv{F_x}{y}) \hat{n}_z
5   \end{equation}

```

$$\operatorname{rot} F = \left(\frac{\partial F_z}{\partial y} - \frac{\partial F_y}{\partial z} \right) \hat{n}_x + \left(\frac{\partial F_x}{\partial z} - \frac{\partial F_z}{\partial x} \right) \hat{n}_y + \left(\frac{\partial F_y}{\partial x} - \frac{\partial F_x}{\partial y} \right) \hat{n}_z \quad (20)$$

Automatic Bracing

When typesetting maths equations, you may use something like `\left(` and `\right)` to make the braces taller than the typical ones. The `physics` package provides some macros to simplify and replace them.

<code>\quantity</code>	<code>\qty(\typical)</code> → 
	<code>\qty(\tall)</code> → 
	<code>\qty(\grande)</code> → 
	<code>\qty[\typical]</code> → 
	<code>\qty \typical </code> → 
	<code>\qty{\typical}</code> → 
	<code>\qty\big{}</code> → 
	<code>\qty\Big{}</code> → 
	<code>\qty\bigg{}</code> → 
	<code>\qty\Bigg{}</code> → 
	<code>\pqty{}</code> ↔ <code>\qty()</code>
	<code>\bqty{}</code> ↔ <code>\qty </code>
	<code>\vqty{}</code> ↔ <code>\qty </code>
	<code>\Bqty{}</code> ↔ <code>\qty{}</code>

automatic `()` braces

automatic `[]` braces

automatic `||` braces

automatic `{ }` braces

manual sizing (works with any of the above bracket types)

alternative syntax; robust and more L^AT_EX-friendly

`\absolutevalue`

$$\backslash\mathrm{abs}\{a\} \rightarrow |a|$$

$$\backslash\mathrm{abs}\Big{a\Big} \rightarrow \Big|a\Big|$$

$$\backslash\mathrm{abs*}\{\mathrm{grande}\} \rightarrow \big| \big|$$

`\norm`

$$\backslash\mathrm{norm}\{a\} \rightarrow \|a\|$$

$$\backslash\mathrm{norm}\Big{a\Big} \rightarrow \Big\|a\Big\|$$

$$\backslash\mathrm{norm*}\{\mathrm{grande}\} \rightarrow \big\| \big\|$$

`\evaluated`

$$\backslash\mathrm{eval}\{x\}_0^{\infty} \rightarrow x \Big|_0^{\infty}$$

$$\backslash\mathrm{eval}(x)_0^{\infty} \rightarrow \left(x\right) \Big|_0^{\infty}$$

$$\backslash\mathrm{eval}[x]_0^{\infty} \rightarrow \left[x\right] \Big|_0^{\infty}$$

$$\backslash\mathrm{eval}[\mathrm{venti}]_0^{\infty} \rightarrow \left[\mathrm{venti}\right] \Big|_0^{\infty}$$

$$\backslash\mathrm{eval}*\{\mathrm{venti}\}_0^{\infty} \rightarrow \left[\mathrm{venti}\right] \Big|_0^{\infty}$$

`\order`

$$\backslash\mathrm{order}\{x^2\} \rightarrow \mathcal{O}\left(x^2\right)$$

$$\backslash\mathrm{order}\Big{x^2\Big} \rightarrow \mathcal{O}\left(x^2\right)$$

$$\backslash\mathrm{order*}\{\mathrm{grande}\} \rightarrow \mathcal{O}\left(\big\| \big\| \right)$$

automatic sizing; equivalent to `\qty`
`|a|`

inherits manual sizing syntax from `\qty`

star for no resize

automatic sizing

manual sizing

star for no resize

vertical bar for evaluation limits

alternate form

alternate form

automatic sizing

star for no resize

order symbol; automatic sizing and space handling

manual sizing

star for no resize

Vector Notation

You may use `\mathbf` to make bold maths symbols, However, it won't always work. For example, with `\mathbf{\alpha}` you may have α , which is actually not bold. These commands will help provide the correct α .

<code>\vectorbold</code>	<code>\vb{a}</code> → \mathbf{a}	upright/no Greek
	<code>\vb*{a}</code> , <code>\vb*{\theta}</code> → \boldsymbol{a} , $\boldsymbol{\theta}$	italic/Greek
<code>\vectorarrow</code>	<code>\va{a}</code> → \vec{a}	upright/no Greek
	<code>\va*{a}</code> , <code>\va*{\theta}</code> → \vec{a} , $\vec{\theta}$	italic/Greek
<code>\vectorunit</code>	<code>\vu{a}</code> → \hat{a}	upright/no Greek
	<code>\vu*{a}</code> , <code>\vu*{\theta}</code> → \hat{a} , $\hat{\theta}$	italic/Greek

There are also some shorthand for vector operations.

<code>\dotproduct</code>	<code>\vdot</code> → \cdot as in $\mathbf{a} \cdot \mathbf{b}$	note: <code>\dp</code> is a protected T _E X primitive
<code>\crossproduct</code>	<code>\cross</code> → \times as in $\mathbf{a} \times \mathbf{b}$	alternate name
	<code>\cp</code> → \times as in $\mathbf{a} \times \mathbf{b}$	shorthand name

The default del (nabla) symbol ∇ used in `physics` vector notation can be switched to appear with an arrow $\vec{\nabla}$ by including the option `arrowdel` in the document preamble \rightarrow `\usepackage[arrowdel]{physics}`.

`\divergence`

`\div` $\rightarrow \nabla \cdot$

note: `amsmath` symbol \div renamed
`\divisionsymbol`
default mode

`\div{\vb{a}}` $\rightarrow \nabla \cdot \mathbf{a}$

`\div(\vb{a}+\tall)` $\rightarrow \nabla \cdot (\mathbf{a} + \text{blue box})$

`\div[\vb{a}+\tall]` $\rightarrow \nabla \cdot \left[\mathbf{a} + \text{blue box} \right]$

long-form

`\curl`

`\curl` $\rightarrow \nabla \times$

`\curl{\vb{a}}` $\rightarrow \nabla \times \mathbf{a}$

`\curl(\vb{a}+\tall)` $\rightarrow \nabla \times (\mathbf{a} + \text{blue box})$

`\curl[\vb{a}+\tall]` $\rightarrow \nabla \times \left[\mathbf{a} + \text{blue box} \right]$

default mode
long-form

`\laplacian`

`\laplacian` $\rightarrow \nabla^2$

`\laplacian{\Psi}` $\rightarrow \nabla^2 \Psi$

`\laplacian(\Psi+\tall)`

$\nabla^2 (\Psi + \text{blue box})$

`\laplacian[\Psi+\tall]` $\rightarrow \nabla^2 \left[\Psi + \text{blue box} \right]$

\rightarrow
default mode
long-form

Operators

The standard set of trig functions is redefined in `physics` to provide automatic braces that behave like `\qty()`. In addition, an optional power argument is provided. This behavior can be switched off by including the option `notrig` in the preamble \rightarrow `\usepackage[notrig]{physics}`.

For example,

<code>\sin</code>	<code>\sin(\grande)</code> \rightarrow $\sin(\blacksquare)$	automatic braces; old <code>\sin</code> renamed <code>\sine</code>
	<code>\sin[2](x)</code> \rightarrow $\sin^2(x)$	optional power
	<code>\sin x</code> \rightarrow $\sin x$	can still use without an argument

Similar behavior has also been extended to the following functions:

<code>\exp(\tall)</code>	<code>exp(\tall)</code>	<code>\exponential</code>
<code>\log(\tall)</code>	<code>log(\tall)</code>	<code>\logarithm</code>
<code>\ln(\tall)</code>	<code>ln(\tall)</code>	<code>\naturallogarithm</code>
<code>\det(\tall)</code>	<code>det(\tall)</code>	<code>\determinant</code>
<code>\Pr(\tall)</code>	<code>Pr(\tall)</code>	<code>\Probability</code>

old definitions \Rightarrow

Use Maths in L^AT_EX

- Math Expressions
- Math Environments
- Spacing in Math Mode
- Basic Math Commands
- Matrices and Arrays

Useful Maths Packages

- Common Packages
- The `physics` Package
- The `systeme` Package
- The `gauss` Package

There are also some new operators:

<code>\trace</code> or <code>\tr</code>	<code>\tr\rho</code> $\rightarrow \operatorname{tr} \rho$ also <code>\tr(\tall)</code> $\rightarrow \operatorname{tr}(\blacksquare)$	trace; same bracing as trig functions
<code>\Trace</code> or <code>\Tr</code>	<code>\Tr\rho</code> $\rightarrow \operatorname{Tr} \rho$	alternate
<code>\rank</code>	<code>\rank M</code> $\rightarrow \operatorname{rank} M$	matrix rank
<code>\erf</code>	<code>\erf(x)</code> $\rightarrow \operatorname{erf}(x)$	Gauss error function
<code>\Res</code>	<code>\Res[f(z)]</code> $\rightarrow \operatorname{Res}[f(z)]$	residue; same bracing as trig functions
<code>\principalvalue</code>	<code>\pv{\int f(z) \dd{z}}</code> $\rightarrow \mathcal{P} \int f(z) \operatorname{d} z$ <code>\PV{\int f(z) \dd{z}}</code> $\rightarrow \operatorname{P.V.} \int f(z) \operatorname{d} z$	Cauchy principal value
<code>\Re</code>	<code>\Re{z}</code> $\rightarrow \operatorname{Re}\{z\}$	old <code>\Re</code> renamed to <code>\real</code> $\rightarrow \Re$
<code>\Im</code>	<code>\Im{z}</code> $\rightarrow \operatorname{Im}\{z\}$	old <code>\Im</code> renamed to <code>\imaginary</code> $\rightarrow \Im$

Quick Quad Text

This set of commands produces text in math-mode padded by `\quad` spacing on either side. This is meant to provide a quick way to insert simple words or phrases in a sequence of equations. Each of the following commands includes a starred version which pads the text only on the right side with `\quad` for use in aligned environments such as `cases`.

`\qqtext`

`\qq{}`

`\qq{word or phrase}` →
 ____ word or phrase ____

general quick quad text with argument
 normal mode; left and right `\quad`

`\qq*{word or phrase}` →
 word or phrase ____

starred mode; right `\quad` only

Some special macros:

`\qcomma` or `\qc` → , ____

right `\quad` only

`\qcc` → ____ c.c. ____

complex conjugate; left and right `\quad` unless starred `\qcc*`
 → c.c. ____

`\qif` → ____ if ____

left and right `\quad` unless starred `\qif*` → if ____

`\qthen`, `\qelse`, `\qotherwise`, `\qunless`, `\qgiven`, `\qusing`, `\qassume`, `\qsince`,
`\qlet`, `\qfor`, `\qall`, `\qeven`, `\qodd`, `\qinteger`, `\qand`, `\qor`, `\qas`, `\qin`

The default differential symbol `d` which is used in `\differential` and `\derivative` can be switched to an italic form *d* by including the option `italicdiff` in the preamble → `\usepackage[italicdiff]{physics}`.

`\differential`

`\dd` → d

`\dd x` → dx

`\dd{x}` → $\mathrm{d}x$

`\dd[3]{x}` → d^3x

`\dd(\cos\theta)` → $d(\cos\theta)$

`\derivative`

`\dv{x}` → $\frac{d}{dx}$

`\dv{f}{x}` → $\frac{df}{dx}$

`\dv[n]{f}{x}` → $\frac{d^n f}{dx^n}$

`\dv{x}(\grande)` → $\frac{d}{dx} \left(\text{blue square} \right)$

`\dv*{f}{x}` → df/dx

- no spacing (not recommended)
- automatic spacing based on neighbors
- optional power
- long-form; automatic braces
- one argument
- two arguments
- optional power
- long-form; automatic braces, spacing
- inline form using `\flatfrac`

Use Maths in L^AT_EX

- Math Expressions
- Math Environments
- Spacing in Math Mode
- Basic Math Commands
- Matrices and Arrays

Useful Maths Packages

- Common Packages
- The `physics` Package
- The `systeme` Package
- The `gauss` Package

<code>\partialderivative</code>	$\partialderivative{x} \rightarrow \frac{\partial}{\partial x}$	alternate name
<code>\pdv{x}</code>	$\pdv{x} \rightarrow \frac{\partial}{\partial x}$	shorthand name
<code>\pdv{f}{x}</code>	$\pdv{f}{x} \rightarrow \frac{\partial f}{\partial x}$	two arguments
<code>\pdv[n]{f}{x}</code>	$\pdv[n]{f}{x} \rightarrow \frac{\partial^n f}{\partial x^n}$	optional power
<code>\pdv{x}(\backslashgrande)</code>	$\pdv{x}(\backslashgrande) \rightarrow \frac{\partial}{\partial x} \left(\text{blue square} \right)$	long-form
<code>\pdv{f}{x}{y}</code>	$\pdv{f}{x}{y} \rightarrow \frac{\partial^2 f}{\partial x \partial y}$	mixed partial
<code>\pdv*{f}{x}</code>	$\pdv*{f}{x} \rightarrow \partial f / \partial x$	inline form using <code>\flatfrac</code>
<code>\variation</code>	$\var{F[g(x)]} \rightarrow \delta F[g(x)]$ $\var{(E-TS)} \rightarrow \delta(E-TS)$	functional variation (works like <code>\dd</code>) long-form
<code>\functionalderivative</code>	$\fdv{g} \rightarrow \frac{\delta}{\delta g}$ $\fdv{F}{g} \rightarrow \frac{\delta F}{\delta g}$ $\fdv{V}{(E-TS)} \rightarrow \frac{\delta}{\delta V} (E-TS)$ $\fdv*{F}{x} \rightarrow \delta F / \delta x$	functional derivative (works like <code>\dv</code>) long-form inline form using <code>\flatfrac</code>

Matrices

The following matrix macros produce unformatted rows and columns of matrix elements for use as separate matrices as well as blocks within larger matrices. For example, the command `\identitymatrix{2}` which also has the shortcut `\imat{2}` produces the elements of a 2×2 identity matrix $\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix}$ without braces or grouping. This allows the command to also be used within another matrix, as in:

Example

```
1 \begin{equation}
2   \begin{pmatrix}
3     \imat{2} \\
4     a & b
5   \end{pmatrix}
6 \end{equation}
```

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \\ a & b \end{pmatrix} \quad (21)$$

To specify elements on the right of left sides of our `\imat{2}` sub-matrix we use the grouping command `\matrixquantity` or `\mqty` to effectively convert `\imat{2}` into a single matrix element of a larger matrix:

Example

```

1 \begin{equation}
2   \begin{pmatrix}
3     \mqty{\imat{2}} & \mqty{a\\b} \\
4     \mqty{c & d}    & e
5   \end{pmatrix}
6 \end{equation}

```

$$\begin{pmatrix} 1 & 0 & a \\ 0 & 1 & b \\ c & d & e \end{pmatrix} \quad (22)$$

The extra `\mqty` groups were required in this case in order to get the a and b elements to behave as a single element, since `\mqty{\imat{2}}` also acts like a single matrix element (the same can be said of the grouped c and d elements). Finally, the outermost `pmatrix` environment could have also been replaced with the `physics` macro `\mqty()`, allowing the above example to be written on one line:

Example

```

1 \begin{equation}
2   \mqty(
3     \mqty{\imat{2}} & \mqty{a\\b} \\
4     \mqty{c & d}    & e
5   )
6 \end{equation}

```

$$\begin{pmatrix} 1 & 0 & a \\ 0 & 1 & b \\ c & d & e \end{pmatrix} \quad (23)$$

The matrix commands are listed below:

<code>\matrixquantity</code>	<code>\mqty{a & b \\ c & d}</code> → $\begin{matrix} a & b \\ c & d \end{matrix}$
	<code>\mqty(a & b \\ c & d)</code> → $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$
	<code>\mqty*(a & b \\ c & d)</code> → $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$
	<code>\mqty[a & b \\ c & d]</code> → $\left[\begin{matrix} a & b \\ c & d \end{matrix} \right]$
	<code>\mqty a & b \\ c & d </code> → $\left \begin{matrix} a & b \\ c & d \end{matrix} \right $
	<code>\pmmqty{} ↔ \mqty()</code>
	<code>\Pmqty{} ↔ \mqty*()</code>
	<code>\bmqty{} ↔ \mqty[]</code>
	<code>\vmqty{} ↔ \mqty </code>
<code>\smallmatrixquantity</code>	<code>\smqty{a & b \\ c & d}</code> → $\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}$
	<code>\smqty()</code> or <code>\spmqty{}</code>
	<code>\smqty*()</code> or <code>\sPmqty{}</code>
	<code>\smqty[]</code> or <code>\sbmqty{}</code>
	<code>\smqty </code> or <code>\svmqty{}</code>
<code>\matrixdeterminant</code>	<code>\mdet{a & b \\ c & d}</code> → $\begin{vmatrix} a & b \\ c & d \end{vmatrix}$
	<code>\smdet{a & b \\ c & d}</code> → $\begin{vmatrix} a & b \\ c & d \end{vmatrix}$
<code>\identitymatrix</code>	<code>\imat{n}</code>
	<code>\smqty(\imat{3})</code> → $\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$

groups a set of matrix elements into a single object

parentheses

alternate parentheses

square brackets

vertical bars

alternative syntax; robust and more L^AT_EX-friendly

the `smallmatrix` form of `\mqty`
small version of `\mqty()`
small version of `\mqty*()`
small version of `\mqty[]`
small version of `\mqty||`

matrix determinant

small matrix determinant
elements of $n \times n$ identity matrix

formatted with `\mqty` or `\smqty`

Use Maths in L^AT_EX

- Math Expressions
- Math Environments
- Spacing in Math Mode
- Basic Math Commands
- Matrices and Arrays

Useful Maths Packages

- Common Packages
- The `physics` Package
- The `systeme` Package
- The `gauss` Package

`\xmatrix`

`\xmat{x}{n}{m}`

`\smqty(\xmat{1}{2}{3})` $\rightarrow \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$
`\smqty(\xmat*{a}{3}{3})` $\rightarrow \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}$
`\smqty(\xmat*{a}{3}{1})` $\rightarrow \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix}$
`\smqty(\xmat*{a}{1}{3})` $\rightarrow \begin{pmatrix} a_1 & a_2 & a_3 \end{pmatrix}$

`\zeromatrix`

`\zmat{n}{m}`

`\smqty(\zmat{2}{2})` $\rightarrow \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$

`\paulimatrix`

`\pmat{n}`

`\smqty(\pmat{0})` $\rightarrow \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$
`\smqty(\pmat{1})` $\rightarrow \begin{pmatrix} 1 & 0 \\ 0 & -i \end{pmatrix}$
`\smqty(\pmat{2})` $\rightarrow \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$
`\smqty(\pmat{3})` $\rightarrow \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$

`\diagonalmatrix`

`\dmat{a,b,c,...}`

`\mqty(\dmat{1,2,3})` $\rightarrow \begin{pmatrix} 1 & & \\ & 2 & \\ & & 3 \end{pmatrix}$

`\mqty(\dmat[0]{1,2})` $\rightarrow \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix}$

`\mqty(\dmat{1,2&3\\4&5})` $\rightarrow \begin{pmatrix} 1 & & \\ & 2 & 3 \\ & 4 & 5 \end{pmatrix}$

`\antidiagonalmatrix` `\admat{a,b,c,...}`

`\mqty(\admat{1,2,3})` $\rightarrow \begin{pmatrix} & & 1 \\ & 2 & \\ 3 & & \end{pmatrix}$

elements of $n \times m$ matrix filled with x

formatted with `\mqty` or `\smqty` star for element indices

as a vector with indices

$n \times m$ matrix filled with zeros equivalent to `\xmat{0}{n}{m}`
 n^{th} Pauli matrix
 $n \in \{0, 1, 2, 3 \text{ or } x, y, z\}$

specify up to eight diagonal or block diagonal elements

optional argument to fill spaces

enter matrix elements for each block as a single diagonal element
same as syntax as `\dmat`

Use Maths in \LaTeX

Math Expressions

Math Environments

Spacing in Math Mode

Basic Math Commands

Matrices and Arrays

Useful Maths Packages

Common Packages

The `physics` Package

The `systeme` Package

The `gauss` Package

1 Use Maths in \LaTeX

2 Useful Maths Packages

- Common Packages
- The `physics` Package
- The `systeme` Package
- The `gauss` Package

The `systeme` Package

To use the `systeme` package, simply insert the command in the preamble of your document.

Command

```
1 \usepackage{systeme}
```

This package can make it really easy when typesetting linear systems by the command `\systeme`.

Example

```
1 \begin{equation}
2   \systeme{
3     2a-3b+4c=2,
4     a+8b+5c=8,
5     -a+2b+c=-5
6   }
7 \end{equation}
```

$$\begin{cases} 2a - 3b + 4c = 2 \\ a + 8b + 5c = 8 \\ -a + 2b + c = -5 \end{cases} \quad (24)$$

It also works for subscripts.

Example

```

1 \begin{equation}
2   \systeme{
3     4x_1-x_2=3,
4     -x_1+5x_2=-1
5   }
6 \end{equation}

```

$$\begin{cases} 4x_1 - x_2 = 3 \\ -x_1 + 5x_2 = -1 \end{cases} \quad (25)$$

It can also reorder the variables and numbers in the equations.

Example

```

1 \begin{equation}
2   \systeme{
3     3y+2x=0,
4     x-z+9=0,
5     2+3x+5-y-7+z=0
6   }
7 \end{equation}

```

$$\begin{cases} 2x + 3y & & = 0 \\ x & - z + & 9 = 0 \\ 3x - & y + z + 2 + 5 - 7 = 0 \end{cases} \quad (26)$$

Complicated coefficients can be handle correctly. Note that + and - should be replaced with `\+` and `\-` in the coefficients.

Example

```

1 \begin{equation}
2   \systeme{
3     (2\+\sqrt{2})x-
4     (1\-\sqrt{2})y=1,
5     x+(1\+\sqrt{2})y=-1
6   }
7 \end{equation}

```

$$\begin{cases} (2 + \sqrt{2})x - (1 - \sqrt{2})y = 1 \\ x + (1 + \sqrt{2})y = -1 \end{cases} \quad (27)$$

The documentation of the `systeme` package can be found in http://mirrors.ctan.org/macros/generic/systeme/systeme_fr.pdf, however it's in French, and the author is Manuel de l'utilisateur.

Use Maths in L^AT_EX

Math Expressions

Math Environments

Spacing in Math Mode

Basic Math Commands

Matrices and Arrays

Useful Maths Packages

Common Packages

The `physics` Package

The `systeme` Package

The `gauss` Package

1 Use Maths in L^AT_EX

2 Useful Maths Packages

- Common Packages
- The `physics` Package
- The `systeme` Package
- The `gauss` Package

The `gauss` Package

To use the `gauss` package, simply insert the command in the preamble of your document.

Command

```
1 \usepackage{gauss}
```

This package provides L^AT_EX-macros for typesetting operations on a matrix. By an “operation on a matrix” we understand a row operation or a column operation. It is named `gauss` because **Gauss Elimination** is a widely used application of matrix operations.

The documentation of the `systeme` package can be found in <http://mirrors.ctan.org/macros/latex/contrib/gauss/gauss-doc.pdf>.

For example, if you are taking VV285 or working with other linear algebra stuffs in L^AT_EX, you may use the `gmatrix` environment provided by the `gauss` package.

Example

```

1  \begin{equation}
2    \begin{array}{ccc|}
3      4 & 2 & -2 & \\
4      -3 & 1 & 0 & \\
5      1 & 4 & 2 & 
6    \end{array}
7    \begin{gmatrix}
8      -2 & 6 & -9
9    \end{gmatrix}
10   \begin{rowops}
11     \swap{0}{2}
12     \add[*]{3}{0}{1}
13   \end{rowops}

```

$$\begin{array}{ccc|c}
 4 & 2 & -2 & -2 \\
 -3 & 1 & 0 & 6 \\
 1 & 4 & 2 & -9
 \end{array}
 \begin{array}{l}
 \leftarrow \begin{array}{l} \leftarrow * (3) \\ \leftarrow + \end{array}
 \end{array}
 \quad (28)$$