

A_MS-L_AT_EX QUICK REFERENCE

Packages

The main package to load is `amsmath`. More symbols are included in `amssymb`.

Typeset

For text style (inline) math, use: `$... $`

For display style math, which breaks the paragraph:
`\begin{equation} ... \end{equation}` (numbered) or
`\[... \]` (non-numbered).

Greek letters

α	<code>\alpha</code>	β	<code>\beta</code>	γ	<code>\gamma</code>
δ	<code>\delta</code>	ϵ	<code>\epsilon</code>	ε	<code>\varepsilon</code>
ζ	<code>\zeta</code>	η	<code>\eta</code>	θ	<code>\theta</code>
ϑ	<code>\vartheta</code>	ι	<code>\iota</code>	κ	<code>\kappa</code>
λ	<code>\lambda</code>	μ	<code>\mu</code>	ν	<code>\nu</code>
ξ	<code>\xi</code>	π	<code>\pi</code>	ϖ	<code>\varpi</code>
ρ	<code>\rho</code>	ϱ	<code>\varrho</code>	σ	<code>\sigma</code>
τ	<code>\tau</code>	υ	<code>\upsilon</code>	ϕ	<code>\phi</code>
φ	<code>\varphi</code>	χ	<code>\chi</code>	ψ	<code>\psi</code>
ω	<code>\omega</code>				
Γ	<code>\Gamma</code>	Δ	<code>\Delta</code>	Θ	<code>\Theta</code>
Λ	<code>\Lambda</code>	Ξ	<code>\Xi</code>	Π	<code>\Pi</code>
Σ	<code>\Sigma</code>	Υ	<code>\Upsilon</code>	Φ	<code>\Phi</code>
Ψ	<code>\Psi</code>	Ω	<code>\Omega</code>		

To ensure a consistent style throughout the document:

```
\renewcommand{\epsilon}{\varepsilon}
\renewcommand{\theta}{\vartheta}
\renewcommand{\rho}{\varrho}
\renewcommand{\phi}{\varphi}
```

Mathematical font

$A B C D E F G H I J K L M N O P Q R S T U V W X Y Z$
`\mathcal{letter}`

Superscript & Subscript

x^y	<code>x^y</code>	x^{a+b}	<code>x^{a+b}</code>
x_y	<code>x_y</code>	x_{a+b}	<code>x_{a+b}</code>

Root

Square root	\sqrt{x}	<code>\sqrt{x}</code>
N-th root	$\sqrt[N]{x}$	<code>\sqrt[N]{x}</code>

Dots

Multiplication dot	\cdot	<code>\cdot</code>	<code>\cdot</code>
Three centered dots	\cdots	<code>\cdots</code>	<code>\cdots</code>
Three baseline dots	\ldots	<code>\ldots</code>	<code>\ldots</code>
Three diagonal dots	\ddots	<code>\ddots</code>	<code>\ddots</code>
Three vertical dots	\vdots	<code>\vdots</code>	<code>\vdots</code>

Spaces

Negative space	<code>\!</code>
Thinnest space	<code>\,</code>
Thin space	<code>\:</code>
Medium space	<code>\;</code>
1em space	<code>\quad</code>
2em space	<code>\qquad</code>

Braces

`\overbrace{ ... }^{text over brace}`
`\underbrace{ ... }_{text under brace}`

Accents

\hat{a}	<code>\hat{a}</code>	\bar{a}	<code>\bar{a}</code>	\mathring{a}	<code>\mathring{a}</code>
\check{a}	<code>\check{a}</code>	\dot{a}	<code>\dot{a}</code>	\vec{a}	<code>\vec{a}</code>
\tilde{a}	<code>\tilde{a}</code>	\ddot{a}	<code>\ddot{a}</code>	\widehat{AAA}	<code>\widehat{AAA}</code>

Operators

<code>\sin</code>	<code>\cos</code>	<code>\arcsin</code>	<code>\arccos</code>	<code>\sinh</code>
<code>\cosh</code>	<code>\tan</code>	<code>\arctan</code>	<code>\log</code>	<code>\ln</code>
<code>\max</code>	<code>\min</code>	<code>\sup</code>	<code>\inf</code>	<code>\tanh</code>
<code>\cot</code>	<code>\sec</code>	<code>\csc</code>	<code>\det</code>	

To define a custom operator:

```
\DeclareMathOperator{\argmax}{argmax}
```

Modulo

$a \bmod b$	<code>a \bmod b</code>
$a \equiv b \pmod{m}$	<code>a \equiv b \pmod{m}</code>

Fractions

`\frac{ ... }{ ... }`

Symbol stacking

`\overset{ ... }{ ... }` `\underset{ ... }{ ... }`

First argument is the main symbol, second argument is the symbol to put over or under the main symbol.

Big operators

$$\int_a^b \quad \int \quad \sum_{k=0}^n \quad \prod_{k=0}^n \quad \lim_{x \rightarrow 0}$$

For multiple integrals: \iint \iiint etc.

Closed path integral: \oint

Delimiter size

Change the delimiter size by adding one of these modifiers immediately before the delimiter itself:

`\big` `\Big` `\bigg` `\Bigg`

Let \LaTeX determine the correct size using `\left` and `\right` immediately before the opening and closing delimiters, respectively.

Absolute value & Norm

$$|x| \quad \lvert x \rvert \quad \|x\| \quad \lVert x \rVert$$

The same can be achieved by defining:

```
\usepackage{mathtools}
\DeclarePairedDelimiter{\abs}{\lvert}{\rvert}
\DeclarePairedDelimiter{\norm}{\lVert}{\rVert}
```

Use starred variants `\abs*` and `\norm*` to produce the correct delimiter height for any kind of equation.

$$|x| \quad \abs{x} \quad \left|\frac{a}{b}\right| \quad \abs*{\frac{a}{b}} \\ \|x\| \quad \norm{x} \quad \left\|\frac{a}{b}\right\| \quad \norm*{\frac{a}{b}}$$

Arrows

$$\begin{array}{llll} \uparrow & \uparrow & \downarrow & \downarrow \\ \Uparrow & \Uparrow & \Downarrow & \Downarrow \\ \leftarrow & \leftarrow & \rightarrow & \rightarrow \\ \leftrightarrow & \leftrightarrow & \Leftarrow & \Leftarrow \\ \Rightarrow & \Rightarrow & \Leftrightarrow & \Leftrightarrow \\ \mapsto & \mapsto & & \end{array}$$

$$\begin{array}{ll} \longleftarrow & \longrightarrow \\ \longleftrightarrow & \Longleftarrow \\ \Longleftarrow & \Longrightarrow \\ \Longleftrightarrow & \Longmapsto \end{array}$$

Binary relations

$$\begin{array}{lll} \neq & \leq & \geq \\ \equiv & \ll & \gg \\ \doteq & \sim & \simeq \\ \subset & \supset & \approx \\ \subseteq & \supseteq & \cong \\ \in & \ni & \propto \\ | & \parallel & \perp \end{array}$$

It's possible to negate these symbols by prefixing them with `\not` (for example: $\not\equiv$)

Binary operators

$$\begin{array}{lll} \pm & \mp & \cdot \\ \div & \times & \setminus \\ \star & \cup & \cap \\ * & \circ & \bullet \\ \oplus & \ominus & \odot \\ \oslash & \otimes & \smallsetminus \end{array}$$

Logic symbols

$$\begin{array}{llll} \vee & \wedge & \neg & \\ \exists & \nexists & \forall & \\ \Rightarrow & \Leftrightarrow & \models & \end{array}$$

Other symbols

$$\begin{array}{lll} \infty & \infty & \\ \partial & \partial & \\ \emptyset & \emptyset & \\ \nabla & \nabla & \\ \langle x \rangle & \langle x \rangle & \end{array}$$

Multi line equations

Use the `multline` environment:

```
\begin{multline} ... \end{multline}
```

To align equations, use the `align` environment.

Specify the alignment position with `&` and separate equations with `\\`:

```
\begin{align}
... &= ... \\
... &= ...
\end{align}
```

Vectors

$$\vec{x} \quad \bm{x}$$

Best practice to easily switch between types:

```
\usepackage{bm}
\renewcommand{\vec}{\bm}
```

Arrays

Use the `array` environment. Use `\\` to separate rows, and `&` to separate elements of each row. To produce large delimiters around the array, use `\left` and `\right` followed by the desired delimiter.

$$\begin{array}{lcl} \left(\begin{array}{lcl} a & b & c \\ d & e & f \\ g & h & i \end{array} \right) \end{array}$$

Each letter in the argument of the array represents a column:

- l left aligned text
- c centered text
- r right aligned text

Cases

Use the `cases` environment. Use `\\` to separate different cases, and `&` for correct alignment.

$$\begin{cases} x & \text{if } x > 0 \\ 0 & \text{if } x \leq 0 \end{cases}$$

Matrices

Use one of the following environments:

<code>matrix</code>	No delimiter
<code>pmatrix</code>	<code>(</code> delimiter
<code>bmatrix</code>	<code>[</code> delimiter
<code>Bmatrix</code>	<code>{</code> delimiter
<code>vmatrix</code>	<code> </code> delimiter
<code>Vmatrix</code>	<code> </code> delimiter

Use `\\` to separate different rows, and `&` to separate elements of each row.

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$$

To produce a small matrix, useful for inline math, use the `smallmatrix` environment: $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$.