

CHAPTER 7: TYPING MATH

NOAH C. RUDERMAN

ABSTRACT. We will review

- Math environments
- Spacing rules
- Equations
- Basic constructs
- Delimiters
- Operators
- Math accents
- Stretchable horizontal lines

A math formula can be typeset inline: $f(x) = y$ or displayed:

$$f(x) = y$$

Actually there are several ways to typeset a formula inline or displayed. The inline formula is part of math environment `math`. We can type it by typing `\(` to begin and `\)` to end the statement. Alternatively we may use `$` signs to begin and end our `math` environment. Just for the benefit of the user, the two methods above for this are shorthand for initializing the actual environment. Thus we can write equations in three ways: $f(x) = y$, or $f(x) = y$, or $f(x) = y$.

According to the book, the most useful way to invoke an inline formula is with the `\(` and `\)` delimiters because \LaTeX has a hard time figuring out incorrect uses of a `$` delimiter since it opens and closes the environment. From now on I will use the `\(` and `\)` delimiters rather than the `$` delimiters.

The `displaymath` environment invokes a displayed formula but obviously the `\[` and `\]` delimiters are more convenient.

Note: no blank *lines* are permitted in a math or displayed math environment. Also, \LaTeX ignores spaces in math.

The following is suggested as good practice for writing math:

- Place `\[` and `\]` on lines by themselves (I already do this)
- Leaves spaces *before* and *after* binary relations
- Indent— by three spaces, for example, the contents of the environment so they stand out

- Keep a formula on a single line of source file, if possible (I don't understand...)

I didn't realize this: the spacing after a comma is different in math and text. Compare $a = b, c$ or d with $a = b, c$ or d . Just... don't include the comma in an inline math formula. I suppose it's debatable if I should go through the effort to do this with a `displaymath` environment.

The `equation` environment and the `displaymath` environment are exactly the same except that the `equation` environment assigns a number to the equation. So yes, the *-ed version of the `equation` environment is equivalent to the `displaymath` environment.

$$(1) \qquad f(x) = y$$

So there is an option `\label` command in the `equation` environment. This allows for easy referencing and can be referenced with the `\ref` command or alternatively with the `\eqref` command. Neither of these commands need to be used in an inline math environment or whatever. For example, see equation (1).

$$a \cdot b \quad a \times b \quad a \div b$$

`\dfrac` types display style fractions inline: $\frac{1+2x}{x+y}$. We can also type inline style fractions in displayed math environments with `\tfrac`.

$$\frac{1+2x}{x+y}$$

I already know how to type superscripts and subscripts: $a^{x^y} - b_{x_y}$. Note that the prime symbol is automatically superscripted and is typed as an apostrophe. $f(x') = y'$. We can also type superscripts and subscripts by themselves as † or $_\dagger$.

Let's type some binomials. We use the `\binom{top expression}{bottom expression}` command.

$$\binom{a}{b+c}$$

We can type display line binomials inline with the `\dbinom` command:

$\dbinom{a}{b+c}$ and we can type inline binomials in displayed math environments with `\tbinom`

$$\tbinom{a}{b+c}$$

There is the low line ellipsis \dots using the command `\ldots` and the on-the-line ellipsis \cdots using the command `\cdots`. \LaTeX decides which one is appropriate if the command `\dots` is used. \LaTeX is not always correct in its judgement so use the more specialized commands if necessary.

This is what a display environment integral looks like

$$\int_{-\infty}^{\infty} x \, dx$$

Notice that the limits aren't really above and below the integral. If we want to make this happen, we use the `\limits` command:

$$\int\limits_{-\infty}^{\infty} x \, dx$$

We can type more than just one integral at a time. Check out the following:

$$\oint \quad \iint \quad \iiint \quad \iiiiiint \quad \int \cdots \int$$

To typeset the square root of a quantity we use the `\sqrt[n root]{argument}` command where *n root* is a number. With this we can typeset

$$\sqrt[3]{\sqrt[4]{x} + 1}$$

Omitting the optional argument typesets the usual square root \sqrt{x} . Sometimes it may be necessary to refine how a root is typeset in the case of $\sqrt[n]{x}$. We can use the command `\leftroot[distance]` where *distance* is an will move the typeset argument left, or right with a negative argument. We can also use the command `\uproot[distance]` where *distance* will move the typeset argument up, or down with a negative statement. Let's try to fix our square root:

$$\sqrt[n]{x}$$

\LaTeX is very fickle about the argument described above. Make sure there is no space between the `[` and the `\` in the command.

We can use the `\text` command in a displayed math environment:

$$a_{\text{left}} + 2 = a_{\text{right}}$$

The `\text` command typesets its argument in size and shape of the surrounding text. There are cases that this might pose a problem. Compare:

The theorem states that a_{right} is ...

Any of the text font commands with arguments can be used in math formulas.

$$() [] \{\} \vee \langle \rangle ||| | [] \uparrow \Uparrow \downarrow \Downarrow \Updownarrow \Updownarrow \lrcorner \lrcorner \llcorner \llcorner$$

All delimiters, except the "corners," can stretch to enclose the typeset formula. For example,

$$\left| \frac{x}{1+y} \right\rangle$$

$$\frac{x}{1+y} \rangle$$

The `\big`, `\Big`, `\bigg`, and `\Bigg` commands produce delimiters of larger sizes that do not stretch.

$$\begin{pmatrix} F(x)|_a^b & F(x)|_a^b & F(x)|_a^b \\ & \begin{pmatrix} \\ \\ \\ \end{pmatrix} & \begin{pmatrix} \\ \\ \\ \end{pmatrix} \end{pmatrix}$$

Delimiters may also be used as binary relations. To elaborate, there are methods to make L^AT_EX recognize the delimiter as a binary operation so there is appropriate spacing on each side. For this there are the `\bigrm`

and `\biggm` commands as well as the other variants.

$$\left\{ x \left| \int_0^x t^2 dt \leq 5 \right. \right\}$$

We can define a new operator with the `\DeclareMathOperator` command. It has the form

`\DeclareMathOperator{\opCommand}{opName}`

and this must be placed in the preamble. Using this we can type `aright` very easily or any other formula like $\frac{x^{x+y}}{\text{weird*moose}}$. This is helpful if we are typing it repeatedly.

Also, here's that weird symbol that I see: \equiv

Some other larger operators used often:

$$\prod_b^a \quad \prod_b^a \quad \sum_b^a \quad \sum_b^a$$

Inline: $\prod_b^a \quad \prod_b^a \quad \sum_b^a \quad \sum_b^a$

Note that the default position for the limits inline is on the side of the operator and the default position of the limits in the displayed formula is on the top and bottom.

It is possible to create multiline subscripts. Use the `\substack` command

$$\sum_{\substack{i < n \\ i \text{ even}}} x_i^2$$

However, the line separator `\\` must always be used to separate lines (but should not be used on the last line).

Here are some math accents that you might use

$\acute{a} \quad \bar{a} \quad \grave{a} \quad \check{a} \quad \dot{a} \quad \ddot{a} \quad \ddot{a} \quad \grave{a} \quad \hat{a} \quad \hat{a} \quad \grave{a} \quad \tilde{a} \quad \tilde{a} \quad \vec{a}$

It is possible to stack the symbols $\hat{\hat{a}}$

Let's create some horizontal braces. We do this with the `\overbrace` command.

$$\overbrace{n_1 + n_2 + n_3 + \cdots}$$

We can add a superscript to this to label the brace

$$\overbrace{n_1 + n_2 + n_3 + \cdots}^{\text{brace label}}$$

The same rules apply to the `\underbrace` command. We can do some pretty fancy things

$$\underbrace{\overbrace{a + \cdots + a}^{\frac{(m-n)}{2}} + \underbrace{b + \cdots + b}_n + \overbrace{a + \cdots + a}^{\frac{(m-n)}{2}}}_m$$

Next we can draw lines above or below a formula with the `\overline` and `\underline` formulas, respectively.

$$\overline{x^2 + y^2 = r^2} \quad \underline{\sin \theta = \frac{h}{r}}$$

We can also place arrows above and below formulas with the `\overleftarrow`, `\overrightarrow`, `\underleftarrow`, `\underrightarrow`, `\overleftrightharrow`, and `\underleftrightharrow` commands.

$$\overleftarrow{aaaa} \quad \underleftarrow{bbbb}$$

Lastly, we can make stretchable arrows to accomodate a formula above or below the arrows with the `\xleftarrow` and `\xrightarrow` commands. The formula on top is given as the required argument (it can be empty) and the formula on the bottom is in the form of an optional argument.

$$A \xrightarrow{1-1} B \xleftarrow[\alpha \rightarrow \beta]{\text{onto}} C \xleftarrow[\gamma]{} D \leftarrow E$$