

Not-So-Frequently Asked Questions for L^AT_EX

Miles

2010

This document addresses more esoteric issues in L^AT_EX that have nonetheless actually arisen with the author. We hope that somebody will find it useful!

- Why is L^AT_EX telling me that a command I use is undefined?

Answer. Make sure that you're using the `amsmath` package. This is included in `rsipacks.sty` (so it will automatically be included in your RSI paper and minipaper). To include it in other documents, put `\usepackage{amsmath}` in your preamble, before `\begin{document}`. Just to be safe, throw in `amssymb` and `amsthm` as well (so that you have all the fonts and symbols you would expect, and so that you can define theorems environments). If the command is a standard one, be sure that you spelled it correctly. If it's a command you defined (or thought you did), make sure that you really defined it.

- How do I get script letters, like \mathcal{L} , \mathcal{H} , \mathcal{F} , and \mathcal{G} ?

Answer. You need the `mathrsfs` package, so put `\usepackage{mathrsfs}` in your preamble. Then you can use the font name `mathscr` to get the desired font in math mode. For example, `\mathscr{L}` yields \mathcal{L} .

- How do I typeset a series of displayed equations so that the equals signs line up?

Answer. First, you need the `amsmath` package (see above). Once you have that, you use the `align*` environment, like this :

```
\begin{align*}
math &= more math \\
&= more math \\
\end{align*}
```

```

other math &\le different math \\
&= yet more math
\end{align*}

```

This will produce something like

$$\begin{aligned}
\sum_{i=1}^n \sum_{j=1}^i f(i, j) &= \sum_{1 \leq j \leq i \leq n} f(i, j) \\
&= \sum_{j=1}^n \sum_{i=j}^n f(i, j); \\
1 + 1 + 1 &= 2 + 1 \\
&= 3.
\end{aligned}$$

You can replace the equals signs with whatever other appropriate symbol you like (\leq , \geq , \equiv , \cong , \subset , etc.). The important things to note are that you must put an `&` before each equals sign (or whatever) that you want aligned, and you need a double backslash (`\\`) whenever you want a new line. If you forget the double backslash, you may get something weird like this :

$$1 + 1 + 1 = 2 + 1 \qquad = 3.$$

- What if I want some/all of my aligned equations numbered?

Answer. Use `align` instead of `align*` to number all of the equations. (In general, `*` in an environment name means “without numbering”.) If you want one particular line not to have a number, put `\notag` on that line any place before the `\\`. Thus if you type something like this :

```

\begin{align}
a &= b \\
&= c \notag \\
&1+2+3+4 \\
&= \notag 3+3+4 \\
&= 6+4 = 10.
\end{align}

```

then you'll get something like this :

$$\begin{aligned} a &= b & (1) \\ &= c \\ 1 + 2 + 3 + 4 &= 3 + 3 + 4 \\ &= 6 + 4 = 10. & (2) \end{aligned}$$

- What if I want a bunch of equations in a row displayed, but not necessarily aligned?

Answer. Use the `gather*` (or `gather`) environment. It works exactly like `align*`, except that the equations aren't aligned, so there's no need for the `&`'s before your equals signs. For example, if you type this :

```
\begin{gather*}
\sum_{k=0}^n 1 = n+1, \\
1+1+1+1+1+1+1 = 7.
\end{gather*}
```

then you'll get this :

$$\begin{aligned} \sum_{k=0}^n 1 &= n + 1, \\ 1 + 1 + 1 + 1 + 1 + 1 + 1 &= 7. \end{aligned}$$

- What if I want to typeset a piecewise-defined function, like this?

$$f(x) = \begin{cases} x^2 - 2x + 1 & \text{if } x \text{ is irrational,} \\ 0, & \text{if } x \text{ is rational.} \end{cases}$$

Answer. You do it like this :

```
\[ f(x) =
\begin{cases}
x^2 - 2x + 1, & \text{if } x \text{ is rational}, \\
0, & \text{if } x \text{ is irrational}.
\end{cases} \]
```

A few things to note here : the double backslash (`\`) tells \LaTeX where your line endings are, as usual. We used the `\text` command so that \LaTeX would interpret the text in text mode. The `\[` and `\]` exist so that this equation is centered and displayed on its own line (and so that it's in math mode). (It's equivalent to `$$. . . $$`.) Finally, the `&`'s are there so that the conditional text lines up. If we took them out, then our output would look like this :

$$f(x) = \begin{cases} x^2 - 2x + 1, & \text{if } x \text{ is rational,} \\ 0, & \text{if } x \text{ is irrational.} \end{cases}$$

Of course, you can have as many different cases as you want. However, like many good things in life, the `cases` environment requires the `amsmath` package.

- Why doesn't \LaTeX like my prime signs?

Answer. The prime symbol (`'`) in math mode is actually shorthand for the sequence `"^{\prime}`". (In math mode, `\prime` gives the prime sign, except that it's enlarged, and not in a superscript, like this : `\prime`.) If you put several primes in a row (as in `a''`), then \LaTeX is okay and knows how to handle it. However, if for some reason you try to use an exponent and then a prime symbol (for instance, `a^{\circ'}`, trying to produce $a^{\circ'}$), then \LaTeX will be unhappy because it thinks you have used a double exponent, and it doesn't like that. You can appease it by replacing your `'` with a `\prime` on the end of your exponent, like this : `a^{\circ \prime}`.

- How can I cram multiple lines into the subscript of a summation symbol?

Answer. If it's really necessary, you can use the `\substack` command, like this :

```
\[ \sum_{\substack{1 \leq p \leq n \\ p \text{ a prime}}} 1 \sim \frac{n}{\log n} . \]
```

That will produce the following output :

$$\sum_{\substack{1 \leq p \leq n \\ p \text{ a prime}}} 1 \sim \frac{n}{\log n}.$$

This command requires the `amsmath` package.

- What if I have a huge equation that won't fit on one line?

Answer. If you only have one equals sign, then you have a few options. One is the `multiline*` environment, which works like `gather*`, except that the first line is aligned on the left, the last line is aligned on the right, and the middle lines are centered. You can also use the `split` environment within other environments; see `amslldoc.pdf` (the documentation for `amsmath`) for more details.

However, if the mess happens in a series of aligned equations, then the following solution is the best I know :

```
\begin{align*}
normal ={}& equation \\
big ={}& stuff + stuff + stuff \\
&+ stuff + more stuff.
\end{align*}
```

We've moved the `&`'s after the equals signs because we want the things right after the equals signs to line up. The `{}` after each `=` is so that the spacing is correct. (If we leave out the `{}`, then the `+` sign is a little farther to the left.) If you type that, you might end up with something like this :

$$\begin{aligned} \det A &= (-1)^{i+1} a_{i1} \det(A_{i1}) + \sum_{k \neq i} (-1)^{k+1} a_{k1} \det(A_{k1}) \\ &= (-1)^{i+1} (b_{i1} + c_{i1}) \det(A_{i1}) \\ &\quad + \sum_{k \neq i} (-1)^{k+1} a_{k1} [\det(B_{k1}) + \det(C_{k1})]. \end{aligned}$$