MATO
- A MARKDOWN TRANSFORMER FRAMEWORK

This little tool is a transformer for markdown formatted text to LaTeX and Groff. It features only a subset of markdown that I commonly use. It tries hard to strike a balance between conformity to the markdown "standard" (whatever that is) and practicality.

I find Markdown to strike a good compromise between syntax getting in the way and power of expression.

Code blocks currently are note supported, but I plan to.

FEATURES

It supports different heading levels (currently only three) and a quotations "style" *.

The groff version `matogro` currently sports a direct compilation mode, which is to say, that it will call pdfmom directly after transforming the markdown source to groff output.

Escaping

Ampersands are special characters in LaTeX, so if you use them in markdown, they have to be escaped in the tex output. Let's try that out: A and B. Let's see, if that also works in teletype expressions: `A and B`. And fat ones: **A and B**. And italics: *A and B*. And fat italics: *A and B*. Is there something like fat italic teletype? `_*A and B*_`. -- It seems there is.

Escaping Differences

As I expanded the output engine to groff, I realized, that there is a challenge: LaTeX and groff see different characters as necessary to escape: in LaTeX it is, amongst others, the Ampersand, in Groff it is a full stop at the beginning of a line!

---

*and even footnotes!

It is written in rust, primarily for me to learn the language, but also for me to learn about writing parsers. * Markdown does not lend itself so well, to write a parser, as it seems to me. Don't try to squeeze anything out of me in the vicinity of the chomsky hiearchy or context freeness.

To tranform a given text in markdown just invoke the `matote` command, like so:

```
matogro <file>
```

---

*To that end, I read the book "Crafting Interpreters" by Robert Nystrom, which I can highly recommend! It is really good and I learned a lot.