Table of contents

1 Contributing 2
1.1 Getting started 2
1.2 Conventions 2
1.2.a Zig 2
1.2.b Docs 2
1.2.c Commits 2

1 Contributing

Thanks for considering contributing to campi-cli.

1.1 Getting started

Enter the dev shell and run the helpers below before opening a pull request.

```
just lint        # treefmt lint
just format      # treefmt format
zig build        # build
zig build test   # run tests
just quarto-render # render the website to check the docs
just quarto-man    # render the man page
```

1.2 Conventions

1.2.a Zig

- Types and structs use PascalCase.
- Constants use UPPER_SNAKE_CASE; `const MARKER: []const u8 = "value";`.
- Variables and function parameters use snake_case, without abbreviations: `index`, `definition`, `argument`, not `i`, `def`, `arg`.
- Functions use camelCase; `resolveFlags`, `findDefinition`, `mergeFlags`.
- Enum variants use ALL_CAPS, union tags use snake_case.
- `|payload|` captures in snake_case.
- The allocator comes first in a function signature; when a method has `self`, `self` comes first, then the allocator.
- Prefer `std.ArrayList` (unmanaged) initialized with `.empty`; pass the allocator to `append`, `toOwnedSlice`, and friends.

1.2.b Docs

- Root markdown doubles as the quarto website content; keep the two in sync.
- No em dashes; use hyphens or semicolons.
- Command listings use inline code followed by `- description`, not tables.
- No mention of integration with platforms that are not yet built.

1.2.c Commits

- Keep changes focused; separate documentation from code where it makes sense.
- Do not commit generated output; `_output/` is gitignored.