# Contributing to Tensor Comprehensions

We want to make contributing to this project as easy and transparent as

possible.

## Pull Requests

We actively welcome your pull requests.

1. Fork the repo and create your branch from `master`.

2. If you've added code that should be tested, add tests.

3. If you've changed APIs, update the documentation.

4. Ensure the test suite passes.

5. Make sure your code lints.

6. If you haven't already, complete the Contributor License Agreement ("CLA").

## Contributor License Agreement ("CLA")

In order to accept your pull request, we need you to submit a CLA. You only need

to do this once to work on any of Facebook's open source projects.

Complete your CLA here: <https://code.facebook.com/cla>

## Issues

We use GitHub issues to track public bugs. Please ensure your description is

clear and has sufficient instructions to be able to reproduce the issue.

Facebook has a [bounty program](https://www.facebook.com/whitehat/) for the safe

disclosure of security bugs. In those cases, please go through the process

outlined on that page and do not file a public issue.

## Coding Style

\* 2 spaces for indentation rather than tabs for C/C++ files

\* 80 character line length

\* Please read [CodingConvenstions.md](CodingConventions.md) file for more conventions

## License

By contributing to TC, you agree that your contributions will be licensed

under the LICENSE file in the root directory of this source tree.

## Automatic formatting

Format your contributed changes with clang-format (using the provided

.clang-format configuration: `clang-format -style=file -i file\_to\_format`).

This can be done automatically by installing the following pre-commit hook

```sh

#!/bin/sh

git diff-index --cached --name-only HEAD | grep -v .md | grep -v third-party | grep -v .txt | grep -v .sh | xargs ./third-party-install/clang+llvm/bin/clang-format -i -style=file

git diff-index --cached --name-only HEAD | xargs git add

```

Paste this code into `./.git/hooks/pre-commit` and run `chmod +x ./.git/hooks/pre-commit`.

\*\*Note:\*\* this will reformat add re-stage all files that you are about to

commit. Even if you cancel the commit (i.e. by supplying an empty commit

message), the staged files will remain formatted.

More info on [git hooks](https://git-scm.com/book/en/v2/Customizing-Git-Git-Hook).

Alternatively, you can use [clang's

git-clang-format.py](https://llvm.org/svn/llvm-project/cfe/trunk/tools/clang-format/git-clang-format)

to format all files that were touched before committing.

Another alternative is to use the provided script:

```

CLANG=${PATH\_TO\_CLANG\_INSTALL\_BINARY}/clang-format ./check\_and\_fix\_format.sh

```

## Changes to isl submodule

If you update the isl submodule, then you will typically also need

to update the C++ bindings in isl\_interface.

To generate these bindings, you need to build with `WITH\_BINDINGS=ON`.