

— Code review —

27th of May – 2nd of June

Organization

You should review the code of another team and provide this review to the other team. The review itself will be evaluated, but it will not affect the overall evaluation of the reviewed team. This task is about mostly about learning how to review code – having your own code reviewed and learning from it is merely a beneficial side effect.

Being reviewed

Please prepare your code for the review until **26th of May**, in particular:

- Mark (currently) unused code as such
- Add proper comments
- Format code (`clang-format -i -style=file <filename>`)

Reviewing

Start by cloning the repository of the team you are reviewing and build their solver. If in doubt ask the other team which strategy to use.

Code reviewing is not only about nicely formatted code with a lot of comments. The following questions may give you an idea what to look for:

- Is the overall architecture reasonable?
- Do all parts of the code make sense to you?
- Can you understand what every function is supposed to do?
- Do the data structures properly support the algorithms?
- Does some code rely on invariants that are not explicitly stated or checked?

Many issues are highlighted by compiler diagnostics. Have a close look at the messages printed while compiling the respective module and possibly have a look at the output of `clang-tidy`. Use the following commands to to this:

```
make smtrat-module-<Module>
make tidy-module-<Module>
```