OpenDP Privacy-Proof Review Board Functioning document

Mission: The mission of the OpenDP Privacy-Proof Review Board (PPRB) is to review, in a timely and fair manner, submissions of proofs contributed by members of the OpenDP community and guarantee that they meet the requirements to be integrated in the OpenDP library. Due to the critical mission of the OpenDP library, code contributions and their proofs need to be vetted, not only to guarantee traditional forms of code quality, but also to guarantee, specifically, that the code contributions satisfy the differential privacy claims that are asserted and justified by the accompanying proofs. Moreover, the contributions will be checked to ensure that they provide the utility/usefulness that the contributors claim.

Organization: The PPRB includes a review board editor in chief, two review board executive editors and six to ten board members. The different roles are specified next.

Editor-in-chief: The main task of the editor-in-chief is to guarantee that the PPRB achieves its mission outlined above. This means that the editor-in-chief's responsibility is to guarantee that contributions are reviewed in a timely manner, avoiding systematic delays, in a fair way, guaranteeing fair consideration and treatment to all the members of the community, and respecting the standards for proof review set by the PPRB. Additional responsibilities of the editor-in-chief include the periodic reassessment of the instructions for reviewers and of the review form. The editor-in-chief is selected by the OpenDP Executive Committee and is expected to commit for a period of four years.

Executive editors: The main task of the executive editors is to organize the work of the PPRB to implement the mission outlined above. This means that the executive editors' responsibilities are identifying members of the editorial board committee, supervising them in the reviewing to guarantee that reviews meet the standard for proof review set by the PPRB, and making sure that the reviewing instructions and reviewing form are up to date with the feedback of the PPRB members. When identifying the members of the editorial board committee, the executive editors will consider technical knowledge as well as diversity considerations. The executive editors are selected by the editor in chief and the previous executive editors. To guarantee continuity, executive editors are expected to commit to their role for a two year rotation: a person would be in this role for two years, the first year with a person who already served, and the second year with a new person.

PPRB members: The main task of the PPRB members is to perform the reviews of the contributions and their proof and express recommendations to the executive editors on whether to accept or reject the proofs. PPRB members are allowed and also encouraged to submit contributions. The reviewing process will be iterative, with potentially multiple rounds of reviews. The PPRB members agree to perform the reviews in a timely and fair manner and to provide formal evidence if the proof/code does not deliver what it is expected to deliver.

Contributors will have the opportunity to review the provided evidence and propose potential fixes. This process will continue until the executive editors have enough information to make a decision on the contribution. The PPRB members are selected by the executive editors and they are expected to have expertise in code review, differential privacy, and programming in general. To guarantee continuity PPRB members are expected to commit to their role for a two year period. During this period, they will be required to review two to four submissions a year - each submission may include multiple proofs.

Reviewing: Reviewers will check the correctness of the pseudocode with respect to the differential privacy claims contained in the proof accompanying the code contribution. To facilitate the work of reviewers, contributors are expected to provide a proof based on the pseudocode, rather than on a mathematical abstraction of their contribution.

Timeline: The PPRB will commit to provide an answer to contributors in one month from the contribution date. Every contribution will be reviewed by two reviewers. The PPRB executive editors reserve the right to extend the reviewing period, or change the numbers of reviews a submission will receive, in special circumstances.

Volunteer for the PPRB: we welcome nomination, including self-nominations, for serving on the proof review board. Participation in the PPRB can provide members valuable insight into the challenges of designing differential privacy contributions and into the process of forming community norms for the contributions to the OpenDP library. If you are interested in participating in this process, or you would like to nominate someone else, please write an email to proof-review@opendp.org with a short description of the expertise of the nominated person.

Submission format:
Submissions are in the form of pull requests (PRs) to the OpenDP Library repository. Here are the details on how to create a submission.

- Clone the OpenDP Library (link).
- Write a LaTex proof (link).
- Open a pull request on the OpenDP Library repository (link, open proof PRs here).

Reviewer Process:
The reviewing process will be iterative, with potentially multiple rounds of reviews. Reviewing will proceed through comments on the pull requests to the OpenDP Library repository. Reviewers post comments on the relevant pull request in the OpenDP Library in order to require clarifications or to point out flaws. Contributors will have the opportunity to answer to the review comments, to clarify aspects that are unclear or to propose potential fixes to flows pointed out by the reviewers. This process will continue until the executive editors have enough information to make a decision on the contribution.

Criteria for Dropping “contrib” Flag on Proof:

- Submission must be meaningful
- Postcondition must follow from the precondition and pseudocode

Criteria for Dropping “floating-point” Flag on Proof:

- Proof accounts for data types used in pseudocode

Criteria for Dropping a Flag on Library Code:

- Proof must not require that flag
- Library code must match the pseudocode

Things to watch out for when reviewing:

- underspecification of some behavior in the pseudocode (e.g. what happens under exceptions like overflow)
- use of correct data types (making sure that data types do not bring implicit violations)
- checking that functions (especially measurements and transformations) are pure: output (or output distribution) should only depend on input, not only any global state. also watch out for side effects.
- underspecified definitions in the Rust documentation (e.g. are dataset metrics for ordered or unordered data)
- not precisely quoting a theorem/lemma from the literature being used