

libqcpp: A C++14 sequence quality control library

Kevin D Murray¹ and Justin O Borevitz¹

¹ ARC Centre of Excellence in Plant Energy Biology, The Australian National University, Canberra, ACT 2602, Australia

DOI: [10.21105/joss.00232](https://doi.org/10.21105/joss.00232)

Software

- [Review](#) ↗
- [Repository](#) ↗
- [Archive](#) ↗

Licence

Authors of JOSS papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC-BY](#)).

Summary

Libqcpp implements a variety of algorithms for Next-generation Sequencing (NGS) data quality control. These algorithms include:

- Sliding-window quality score trimming, using an algorithm based on Sickel (Joshi and Fass 2011).
- A combined adaptor removal and read merging algorithm for paired end reads that uses global pairwise alignment of reads. This algorithm is similar to AdapterRemoval (Lindgreen 2012).
- Cycle-wise summarisation of base quality scores, similar to FastQC (Andrews 2012)

Libqcpp allows simple composition of quality control pipelines that combine these features into a single unit. Application code can then simply read from a stream of sequence reads that have passed quality control measures. Optionally, parsing and quality control can occur in one or more background threads for efficiency. Reports detailing actions performed and summaries of results may be obtained in YAML format. Libqcpp includes `trimit`, a command line interface to these features for those not building their own applications.

Libqcpp uses the SeqAn library for sequence parsing and alignment (Döring et al. 2008), libyaml-cpp for YAML report generation, and Catch for unit testing. Documentation on API and command line usage is included, and available at <https://qcpp.readthedocs.io/>.

References

- Andrews, S. 2012. “FastQC A Quality Control Tool for High Throughput Sequence Data.” <http://www.bioinformatics.babraham.ac.uk/projects/fastqc/>.
- Döring, Andreas, David Weese, Tobias Rausch, and Knut Reinert. 2008. “SeqAn an Efficient, Generic C++ Library for Sequence Analysis.” *BMC Bioinformatics* 9: 11. doi:10.1186/1471-2105-9-11.
- Joshi, N A, and J N Fass. 2011. *Sickle: A Sliding-Window, Adaptive, Quality-Based Trimming Tool for FastQ Files* (version 1.33). <https://github.com/najoshi/sickle>.
- Lindgreen, Stinus. 2012. “AdapterRemoval: Easy Cleaning of Next-Generation Sequencing Reads.” *BMC Research Notes* 5: 337. doi:10.1186/1756-0500-5-337.