General improvements to the RAMTools bioinformatics suite

Contact information:

Mentor: Vassil Vassilev

Email: Vassil.Vassilev@cern.ch

Website: vassil.vassilev.info

Student: Georgi Haralanov

Email

-School: georgiharalanov_e23@schoolmath.eu

-Personal: georgiharalanov2009@gmail.com

Introduction

My name is Georgi Haralanov and as part of my school curriculum I've decided to join the development of the RAMTools project under the mentorship of the Compiler Research Group. I'm glad to be able to work with professionals in the field of CS such as Vassil Vassilev on such a project.

The problem of increasing data

During the last couple of years the amount of data produced by genomic sequencing tasks has increased by an astronomical amount. Based on estimates the current data stored has overtaken the collective data stored on social media. The price of sequencing a genome is halved approximately every 5 months while the price of storage solutions drops every 14 months. Thus it is evident that the speed of data collection far outweighs our capacity to contain it.

Possible solutions:

Using 2-bit encoding exploiting ROOT's RNTuple format and compressing quality score statistics will allow us to reduce the amount of data used compared to plaintext or older filetypes which use 8-bit encoding. It is also possible to employ a multi-compression strategy which includes stacking the effects of more than one compression algorithm. This shows diminishing effects since data has a compression limit before encountering loss of information. This allows for faster transmission of data but not for faster processing since the files will require multiple decompression cycles.

Speed problems

Searching through a couple of GBs might take a couple of seconds or more depending on the hardware which is used. That is an acceptable time loss for small scale projects or per person data analysis but taking into account the petabyte or exabyte scale of some of the larger analysis and research projects which require searching through sample data of a city's or region's population it is simply not viable to wait that long for query results. RNTuple files have already been proven to be as fast as CRAM files when using medium sized data portions and faster when querying when using large files.

Possible solutions:

Currently Ramtools utilizes single-core execution. While the speed gained just by efficient search algorithms has been extraordinary we can continue this trend by allowing the use of ROOT's already existing multithreaded i/o infrastructure. By splitting the file into manageable chunks and allowing several cores to work on them in parallel will lead to a several times increase in speed.

Project Timeline

Week	Description	Deliverables
1-2	Familiarize myself with the ROOT ecosystem and the direction of development in RAMTools	Benchmarks comparing current speeds of Ramtools with samtools and previous tests
3-4	Explore optimization possibilities in search algorithms and implement any possible improvements	A list of possible ways to improve speeds or memory usage
5	Implement search algorithm optimization	Generated code with improved efficiency
6	Test improved code for any bugs or issues and fixes for anything found	Documentation on findings of tests and fixed errors

7	Explore ROOT's Implicit Multithreading (IMT)	Generated code with improved speed
8-9	Implement multithreading in Ramtools using IMT	
10	Test multithreaded implementation for any issues and bugs	Fixes for issues and documentation on found issues
11	Benchmark parallel vs non-parallel speeds	Benchmarks with speed differences for parallelism in Ramtools
12	Compile findings into concrete documentation	Complete statistics for all work done and results so far

Feasibility and time frame

The possible improvements explored in the earlier parts of the project proposition and the time frame given are not final. It is possible that some proposed changes do not offer satisfactory results and as such shall be discarded during production. The time table described above concerns the time frame from Feb 2026 to June 2026. Work on the project will continue until September 2026 using the free time afforded to me during the summer break.

Citations:

https://pmc.ncbi.nlm.nih.gov/articles/PMC4494865/ (genomic data growth and statistics)

https://www.istc-cc.cmu.edu/publications/papers/2013/EECS-2013-207.pdf (research showing increase in performance using more complex file formats)

https://pmc.ncbi.nlm.nih.gov/articles/PMC10512525/ (description and comparison showing different tools for parsing and querying genomic files)

https://compiler-research.org/assets/docs/AdityaPandey_GeneROOT2026.pdf (current timeline for project advancement)

https://compiler-research.org/assets/presentations/Aditya_Pandey_GSoC2025_final.pdf (speed and size comparison between cram and root files)