Prime Implicants for Quine-McCluskey

Boolean functions are maps of the form $f: \{0,1\}^n \to \{0,1\}$, where n is a non-negative integer. The goal of this project is to minimize a given Boolean function by using the algorithm by Quine [1, 2] and McCluskey [3]. The algorithm consists of two stages:

1. First, all prime implicants (maximal subset of literals that imply the function) are determined.
2. Second, a minimum subset of the prime implicants that covers the entire function is computed.

Both steps are known to be NP-complete [4]. Wikipedia provides a detailed example as well as pseudocode [5]. A more efficient implementation can be found on GitHub [6]. It can be used as a benchmark and provide you with some ideas (you should still write your own code).

In this project, your goal will be to speed up *only* the *first* step. To this end:

1) Implement a basic version in C and verify its correctness by using unit tests and comparing to other implementations. Your implementation should have time complexity $\Theta(n3^n)$.
2) Write a high-performance version in C using the techniques taught in the lecture. You are allowed to change the representation of data and rearrange computation. You may assume that you know the number of variables at compile time (i.e., code generation is allowed). You should not be able to improve on the asymptotic complexity (otherwise consult with supervisor).
3) **Optional**: implement a version that is memory-efficient and can thus solve larger instances with a limited amount of RAM (but is potentially slower).

References

[1]: Quine, W. V. (1952). The Problem of Simplifying Truth Functions. *The American Mathematical Monthly*, *59*(8), 521–531. https://doi.org/10.1080/00029890.1952.11988183

[2]: Quine, W. V. (1955). A Way to Simplify Truth Functions. *The American Mathematical Monthly*, *62*(9), 627–631. https://doi.org/10.1080/00029890.1955.11988710

[3]: E. J. McCluskey, "Minimization of Boolean functions," in The Bell System Technical Journal, vol. 35, no. 6, pp. 1417-1444, Nov. 1956, doi: https://doi.org/10.1002/j.1538-7305.1956.tb03835.x.

[4]: C. Umans, T. Villa and A. L. Sangiovanni-Vincentelli, "Complexity of two-level logic minimization," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 25, no. 7, pp. 1230-1246, July 2006, doi: https://doi.org/10.1109/TCAD.2005.855944.

[5]: https://en.wikipedia.org/wiki/Quine%E2%80%93McCluskey_algorithm

[6]: https://github.com/hellman/Quine-McCluskey