

Quine-McCluskey

Quine-McCluskey method is an algorithm that minimizes Boolean expressions **without limiting the number of input variables unlike the Karnaugh map**. The automation of the Quine-McCluskey algorithm is also quite easy to do.

[Term project] Implement the Quine-McCluskey algorithm in C/C++ for a Boolean function with any number of inputs larger than zero. The input and output file formats are shown below.

Input format	Output format
File name: input_minterm.txt	File name: result.txt
4 // input bit length d 0000 // don't care value m 0100 // input having the result with true m 0101 m 0110 m 1001 m 1010 d 0111 d 1101 d 1111	01-- 1-01 1010 Cost (# of transistors): 40

Cost of a SOP expression

Q-M algorithm computes a SOP expression of a function. A SOP expression can be implemented by a two-level logic circuit composed of inverters, AND gates, and an OR gate. Let the *cost* of a SOP expression be the *number of transistors* needed for the two-level logic circuit implementation of the SOP.

For example, $A'BC' + C'DEF$ needs two inverters (4 transistors), one three-input AND gate (8 tr.), one four-input AND gate (10 tr.), and one two-input OR gate (6 tr.). The cost of $A'BC' + C'DEF$ is 28.

What you need to submit:

- Report
 - Problem statement (15%)
 - Your algorithm with pseudo code and flow chart (30%)
 - Verification strategy & corresponding examples with explanation (50%)
 - A testcase that you think it is very hard to solve (5%)
 - ✓ This might be used to verify the results of classmates.
- Program source code with sufficient comments (No English comment allowed except any terminologies)
- Project file **except .vs folder** (Visual studio version: above 2019)
- Examples that you used to verify your program (The quantity as well as the quality of examples are very important).