Quine-McCluskey Calculator

User Manual

Caquilala, Cedric
Famadico, Nichol John

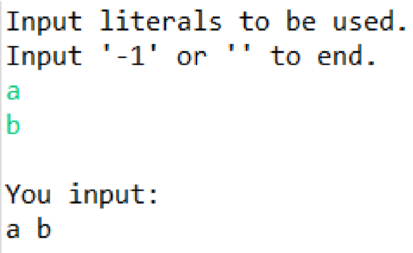
Overview

This program is a freeware to minimize Boolean functions given its minterms by employing the Quine-McCluskey algorithm.

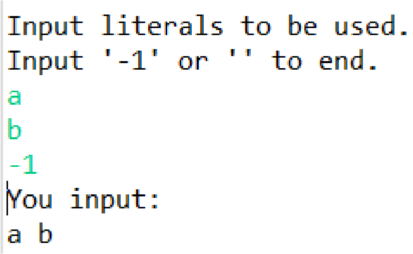
The Quine-McCluskey algorithm was developed by Williard Quin and Edward McCluskey sometime between 1952 to 1956. It was designed to find the prime implicants of the function composed of minterms. It is very similar to the Karnaugh Map, but it utilizes tables instead of grids to make the implementation into computer programs easier.

Getting Started

This program is console-based. Users can submit their input by typing and pressing the return key. To move on to the next step, the user can input '-1', or simply press the return key without any input.



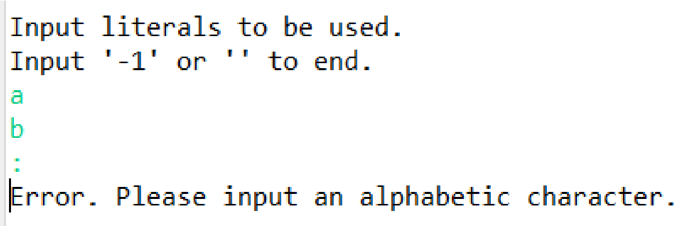
```
Input literals to be used.
Input '-1' or '' to end.
a
b

You input:
a b
```



```
Input literals to be used.
Input '-1' or '' to end.
a
b
-1
You input:
a b
```

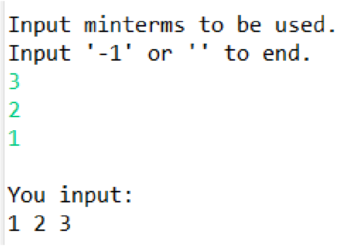
Inputting Data

The first input that will be asked from the user is the literals that will be used. This will determine what variables will be displayed once the calculations are finished as well as the number of bits that will be used. Literals must be alphabetic, with no repetition. There is no limit to the maximum number of literals.

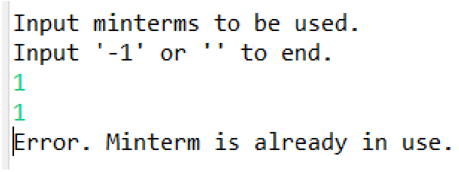


```
Input literals to be used.
Input '-1' or '' to end.
a
b
:
Error. Please input an alphabetic character.
```

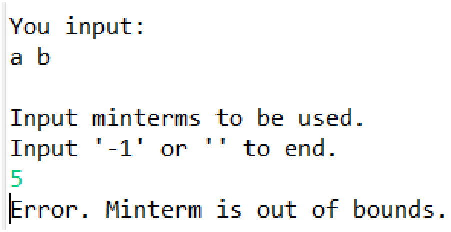
The next input will be the minterms. Minterms must be integers ranging from 0 up to the maximum value allowed by bits (2^n). No repetition is allowed. Ordering the minterms is not necessary, as the program will sort is in ascending order automatically during calculation.



```
Input minterms to be used.
Input '-1' or '' to end.
3
2
1

You input:
1 2 3
```



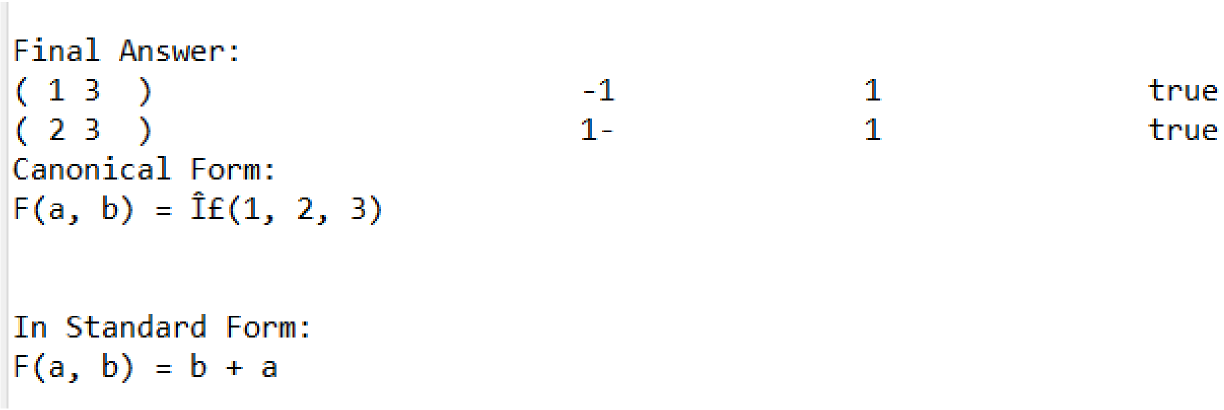
```
Input minterms to be used.
Input '-1' or '' to end.
1
1
Error. Minterm is already in use.
```



```
You input:
a b

Input minterms to be used.
Input '-1' or '' to end.
5
Error. Minterm is out of bounds.
```

Final Answer

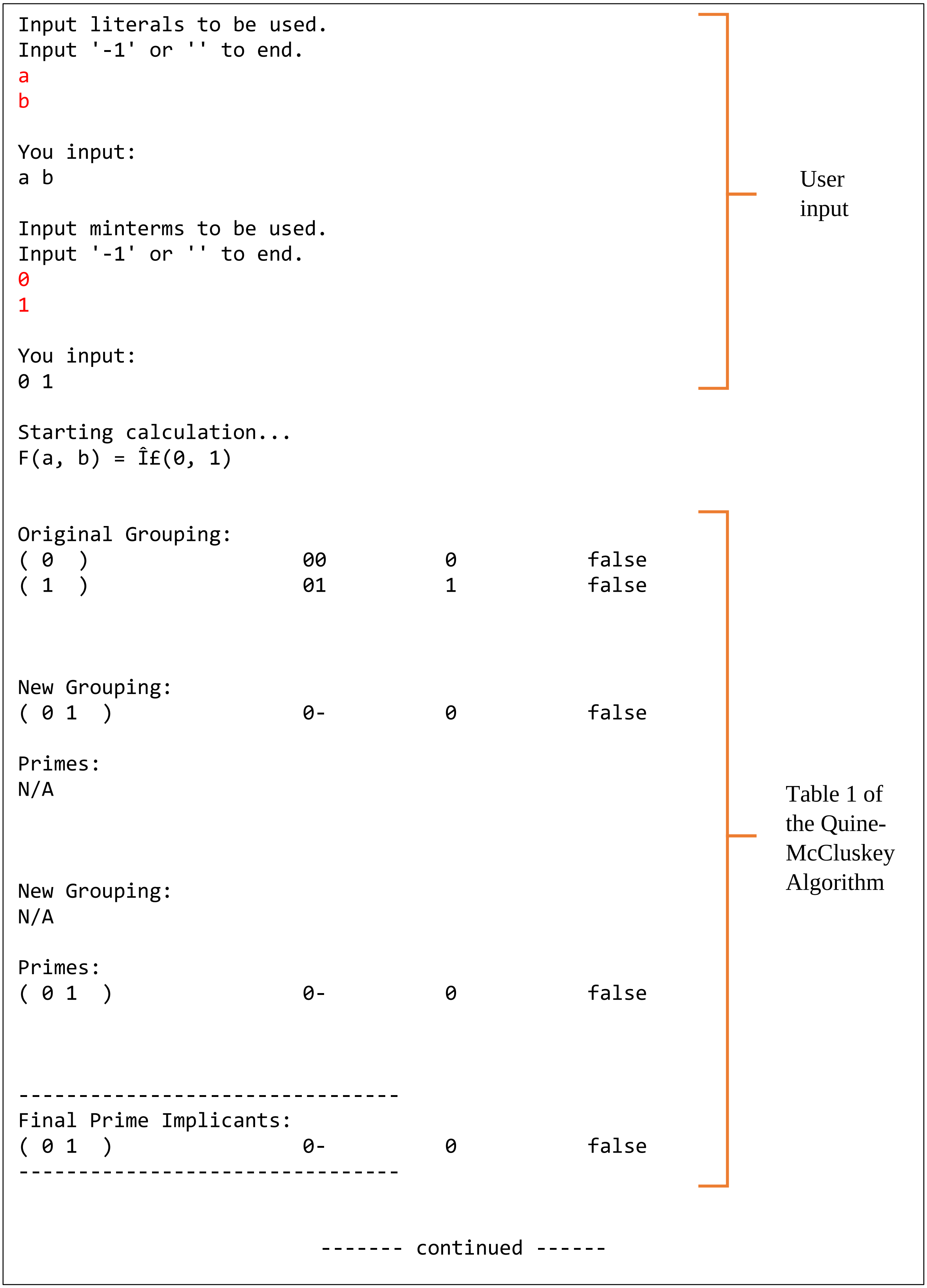
The final answer will be displayed after inputting the minterms. The program will return the simplified form in its standard form using the given literals at the beginning.



```

Final Answer:
( 1 3  )                  -1           1            true
( 2 3  )                  1-           1            true
Canonical Form:
F(a, b) = Î£(1, 2, 3)


In Standard Form:
F(a, b) = b + a
```

In addition, a detailed solution is printed before the final answer. It is printed right after input and before the final answer. The solution is divided into Table 1 and Table 2 of the Quine-McCluskey method.

An example of use is as follows:



```
Input literals to be used.
Input '-1' or '' to end.
a
b

You input:
a b

Input minterms to be used.
Input '-1' or '' to end.
0
1

You input:
0 1

Starting calculation...
F(a, b) = Î£(0, 1)


Original Grouping:
( 0  )                  00          0           false
( 1  )                  01          1           false



New Grouping:
( 0 1  )                0-          0           false

Primes:
N/A



New Grouping:
N/A

Primes:
( 0 1  )                0-          0           false




--------------------------------
Final Prime Implicants:
( 0 1  )                0-          0           false
--------------------------------


                         ------- continued ------
```

```
                    ------- continued ------



Table 2
0     1
x     x     [0, 1]

/     /


Final Answer:
( 0 1  )                0-          0           true
Canonical Form:
F(a, b) = Î£(0, 1)


In Standard Form:
F(a, b) = a'




                       ------- end ------
```

Table 2 of the Quine-McCluskey Algorithm

Final Answers

References:

McCluskey, E. Jr., (November 1956). "Minimization of Boolean Functions". Bell System Technical Journal. 35 (6): 1417–1444. doi:10.1002/j.1538-7305.1956.tb03835.x.

Quine, W. (1952). "The Problem of Simplifying Truth Functions". The American Mathematical Monthly. 59 (8): 521–531. doi:10.2307/2308219. JSTOR 2308219.