

1. Installation and running

The demo executable file is named `m8_demo.exe`.

The file `meth8_parameter_file.txt` is required to be located in the same directory as is the demo executable to instruct the demo which logic system to use. The default for the demo is system "VL4".

The input file for equations is named `meth8_input_file.txt`. The format of an equation to test ends in a semi-colon as the terminating character, otherwise an exception is raised.

The output file name is unique with an embedded time stamp and not overwritten. The final truth table value result is at the bottom of the file after results for each step. If Model 1 is of most interest, the output file can be made less unwieldy by searching and replacing instances of the characters U, I, P, E with null.

2. Usage tips

If an expression is well formed, sometimes it parses with an exception raised. This results from pairs of operators preceding variables *only* and not expressions. If this happens, we suggest a work around by embedding the expression as an antecedent (or consequent) with a connective to tautology such as " $p=p$ ". T tautology is the designated proof value, F contradiction, N for truthity, and C for falsity.

3-8. Parsing anomalies

The truth tables of 16-values are presented horizontally as row-major. We found these examples with work arounds.

- 3.1 $\#(\sim(\% \sim p \> \sim \% r)) > \#(\sim(\# r \> \% p))$; `ngp` (not good parse)
 3.2 $\#(\sim(\% \sim p \> \sim \% r)) > \#(\sim(\# r \> \% p)=(p=p))$; Injecting equality on the antecedent yields tautology (proof).
 3.3 $\#(\sim(\% \sim p \> \sim \% r)) > \sim(\# r \> \% p)=(p=p)$; *Not* distributing the modal operator yields non-contingency (truthity).

- 4.1 $\# \sim(\% \sim p \> \sim \% r) > \# \sim(\# r \> \% p)$; `ngp`
 4.3 $\#(\sim(\% \sim p \> \sim \% r)) > \#(\sim(\# r \> \% p)=(p=p))$; This matches Eq. 3.3.

- 5.1 $(\sim(\#(\% \sim p \> \sim \% r))) > (\sim(\#(\# r \> \% p)))$; `ngp`
 5.2 $(\sim(\#(\% \sim p \> \sim \% r)) > \#(\# r \> \% p))=(p=p)$; Distributing negation and injecting equality yields contradiction.

- 6.1 $\sim(\#(\% \sim p \> \sim \% r)) > \sim(\#(\# r \> \% p))$; `ngp`
 6.2 $\sim(\#(\% \sim p \> \sim \% r)) > \#(\# r \> \% p)=(p=p)$; Injecting equality to force an antecedent yields contradiction.

- 7.1 $\sim \#(\% \sim p \> \sim \% r) > \sim \#(\# r \> \% p)$; `ngp`, matching 6.1.
 7.2 $\sim(\#(\% \sim p \> \sim \% r)) > \#(\# r \> \% p)=(p=p)$; This matches 6.2.

- 8.1 $\#(\% \sim p \> \sim \% r) > \#(\# r \> \% p)$; Validated as tautology (proof).
 8.2 $\#((\% \sim p \> \sim \% r) > (\# r \> \% p))=(p=p)$; *Not* distributing the quantifier yields non-contingency (truthity).

9. Support

We are available for questions from 9 AM – 5 PM MT on M-F at (719) 210-9534 or by email at info@ersatz-systems.com.