Motivation

```
Algorithm: WhatDoIDo(n, m)
Input  : Two positive integers n, m.
Output: The number contained in n.
while (m > 0) do
    m = m -1 ;
    n = n + 1;
end
return n;
```

In First-Order Logic Modulo LIA

2	$\forall n, m.$	$(m > 0, R(2, n, m)$	$\rightarrow$	$R(3, n, m))$
2	$\forall n, m.$	$(m = 0, R(2, n, m)$	$\rightarrow$	$R(6, n, m))$
3	$\forall n, m, m'.$	$(m' = m - 1, R(3, n, m)$	$\rightarrow$	$R(4, n, m'))$
4	$\forall n, m, n'.$	$(n' = n + 1, R(4, n, m)$	$\rightarrow$	$R(5, n', m))$
5	$\forall n, m.$	$(R(5, n, m)$	$\rightarrow$	$R(2, n, m))$

$$\forall n, m \,.\, (R(2, n, m) \rightarrow R(6, n + m, 0))$$

2-Counter Machines (Minsky 1967)

The memory of the machine are two integer counters k_1, k_2, where the integers are not limited in size, resulting in the name. The counters may be initialized at the beginning with arbitrary positive values.
A program consists of a finite number of programming lines, each coming with a unique and consecutive line number and containing exactly one instruction. The available instructions are:

inc(k_i)	increment counter k_i and goto the next line,
td(k_i, n)	if $k_i > 0$ then decrement k_i and goto the next line, otherwise goto line n and leave counters unchanged,
goto n	goto line n,
halt	halt the computation.

Example: WhatDoIDo

2 $\mathrm{td}(k_2, 6)$
4 $\mathrm{inc}(k_1)$
5 goto 2
6 halt

8.7.1 Theorem (2-Counter Machine Halting Problem)

The halting problem for 2-counter machines is undecidable (Minsky 1967).

Proof.

(Idea) By a reduction to the halting problem for Turing machines. □

8.7.2 Proposition (FOL(LIA) Undecidability with a Single Ternary Predicate)

Unsatisfiability of a FOL(LIA) clause set with a single ternary predicate is undecidable.

FOL(LIA) Decidable for Binary or Monadic Predicates?

No: translate 2-counter machine halting problem to FOL(LIA) with a single monadic predicate.

Idea: translate state (i, n, m) where the program is at line i with respective counter values n, m by the integer $2^n \cdot 3^m \cdot p_i$ where p_i is the i^{th} prime number following 3

Example: WhatDoIDo

1 td(k_2, 4)
2 inc(k_1)
3 goto 1
4 halt

$$
\begin{aligned}
5y = x, 3y' = y, x' = 7y', S(x) &\to S(x') \\
5y = x, 3y' + 1 = y, x' = 13y', S(x) &\to S(x') \\
5y = x, 3y' + 2 = y, x' = 13y', S(x) &\to S(x') \\
7y = x, x' = 2y, x'' = 11x', S(x) &\to S(x'') \\
11y = x, x' = 5y, S(x) &\to S(x') \\
13y = x, S(x) &\to
\end{aligned}
$$

8.7.3 Proposition (FOL(LIA) Undecidability with a Single Monadic Predicate)

Unsatisfiability of a FOL(LIA) clause set with a single monadic predicate is undecidable (Downey 1972).