This document contains the program traces and explanations of the execution of the InduceNFA procedure presented in the paper entitled *A Comparison of Selected Variable Ordering Methods for NFA Induction.* The paper was written by Tomasz Jastrząb and submitted to the International Conference on Computational Science (ICCS 2019).

Let $S = (\{a, abb\}, \{aaa, ab, ba\})$ be the input sample over the alphabet $\Sigma = \{a, b\}$. Let the number of states $k = 2$, and let $y_1 = 0$ and $y_2 = 1$. Let the set of constraints $C = \{c_1, c_2, c_3, c_4, c_5\}$ be as follows, with constraints c_1 and c_2 corresponding to examples, and constraints c_3–c_5 corresponding to counterexamples:

$$\begin{array}{lll}
\mathbf{c_1}: & x_1y_1 + x_2y_2 & = 1 \\
\mathbf{c_2}: & (x_1x_5^2 + x_1x_6x_7 + x_2x_5x_7 + x_2x_7x_8)y_1 + (x_1x_5x_6 + x_1x_6x_8 + x_2x_6x_7 + x_2x_8^2)y_2 & = 1 \\
\mathbf{c_3}: & (x_1^3 + 2x_1x_2x_3 + x_2x_3x_4)y_1 + (x_1^2x_2 + x_1x_2x_4 + x_2^2x_3 + x_2x_4^2)y_2 & = 0 \\
\mathbf{c_4}: & (x_1x_5 + x_2x_7)y_1 + (x_1x_6 + x_2x_8)y_2 & = 0 \\
\mathbf{c_5}: & (x_1x_5 + x_3x_6)y_1 + (x_2x_5 + x_4x_6)y_2 & = 0
\end{array}$$

Since it holds that $y_1 = 0$ and $y_2 = 1$ the constraints can be simplified as follows:

$$\begin{array}{lll}
\mathbf{c_1}: & x_2 & = 1 \\
\mathbf{c_2}: & x_1x_5x_6 + x_1x_6x_8 + x_2x_6x_7 + x_2x_8^2 & = 1 \\
\mathbf{c_3}: & x_1^2x_2 + x_1x_2x_4 + x_2^2x_3 + x_2x_4^2 & = 0 \\
\mathbf{c_4}: & x_1x_6 + x_2x_8 & = 0 \\
\mathbf{c_5}: & x_2x_5 + x_4x_6 & = 0
\end{array}$$

Given the above constraints the numbers of active products in constraints c_1–c_5 are $|c_1| = 1$, $|c_2| = 4$, $|c_3| = 4$, $|c_4| = 2$, $|c_5| = 2$. The number of active products $d(c, x_i)$ involving variables x_i, for $i = 1, 2, \ldots, 8$, are as follows:

1. For c_1: $d(c_1, x_1) = 0$, $d(c_1, x_2) = 1$, $d(c_1, x_3) = 0$, $d(c_1, x_4) = 0$, $d(c_1, x_5) = 0$, $d(c_1, x_6) = 0$, $d(c_1, x_7) = 0$, $d(c_1, x_8) = 0$,
2. For c_2: $d(c_2, x_1) = 2$, $d(c_2, x_2) = 2$, $d(c_2, x_3) = 0$, $d(c_2, x_4) = 0$, $d(c_2, x_5) = 1$, $d(c_2, x_6) = 3$, $d(c_2, x_7) = 1$, $d(c_2, x_8) = 2$,
3. For c_3: $d(c_2, x_1) = 2$, $d(c_2, x_2) = 4$, $d(c_2, x_3) = 1$, $d(c_2, x_4) = 2$, $d(c_2, x_5) = 0$, $d(c_2, x_6) = 0$, $d(c_2, x_7) = 0$, $d(c_2, x_8) = 0$,
4. For c_4: $d(c_2, x_1) = 1$, $d(c_2, x_2) = 1$, $d(c_2, x_3) = 0$, $d(c_2, x_4) = 0$, $d(c_2, x_5) = 0$, $d(c_2, x_6) = 1$, $d(c_2, x_7) = 0$, $d(c_2, x_8) = 1$,
5. For c_5: $d(c_2, x_1) = 0$, $d(c_2, x_2) = 1$, $d(c_2, x_3) = 0$, $d(c_2, x_4) = 1$, $d(c_2, x_5) = 1$, $d(c_2, x_6) = 1$, $d(c_2, x_7) = 0$, $d(c_2, x_8) = 0$.

In what follows we present the operations performed in the evaluation and ordering phases of the InduceNFA procedure at the successive levels of recursion. While specifying each recursion level we also provide the values of the procedure arguments, namely variables i and v representing the index and value of the most recently set variable x.

1 The *min-max-ex* variable ordering method

Level 0, $i = \emptyset$, $v = \emptyset$. At first the Reorder procedure is called. The constraint related to examples having the fewest active products is c_1. The constraint contains only a single variable x_2, so this variable is selected as the next variable. Therefore, $i \leftarrow 2$ and $v \leftarrow 0$, as the Reorder procedure always assigs zero first (see row 1 in Tab. 1).

Level 1, $i = 2$, $v = 0$. Since $v = 0$, the EVALUATE procedure updates the values $|c|$ and $d(c, x_i)$ in the following way: $|c_1| \leftarrow 0$, $|c_2| \leftarrow 2$, $d(c_1, x_2) \leftarrow 0$, $d(c_2, x_2) \leftarrow 0$, $d(c_2, x_6) \leftarrow 2$, $d(c_2, x_7) \leftarrow 0$, $d(c_2, x_8) \leftarrow 1$ (products x_2, $x_2x_6x_7$, and $x_2x_8^2$ become inactive). Next the conditions $|c_1| = 0$ and $|c_2| = 0$ are checked and since the first condition is satisfied, we set $p \leftarrow$ **false** and return. As a result, we backtrack to level 0, change the value of variable $x_2 \leftarrow 1$ and invoke INDUCENFA again.

Level 1, $i = 2$, $v = 1$. The EVALUATE procedure now checks whether any of the constraints c_3–c_5 contains a product equal to one. As there is no such product, we continue to check whether constraints c_1 or c_2 contain a product equal to one. We find that c_1 becomes satisfied, but since c_2 is not yet satisfied we set $p \leftarrow$ **true** and return (see row 2 in Tab. 1). The REORDER procedure chooses from c_2 (the only unsatisfied constraint related to examples), the most frequent variable x_6.

Level 2, $i = 6$, $v = 0$. The EVALUATE procedure updates the values $|c_2|$ and $d(c_2, x_i)$ as follows: $|c_2| \leftarrow 1$, $d(c_2, x_1) \leftarrow 0$, $d(c_2, x_5) \leftarrow 0$, $d(c_2, x_6) \leftarrow 0$, $d(c_2, x_7) \leftarrow 0$, $d(c_2, x_8) \leftarrow 1$ (products $x_1x_5x_6$, $x_1x_6x_8$, and $x_2x_6x_7$ become inactive). Since $|c_2| \neq 0$ we set $p \leftarrow$ **true** and proceed (see row 3 in Tab. 1). The REORDER procedure selects now the only remaining variable in c_2, namely x_8 and sets it to zero.

Level 3, $i = 8$, $v = 0$. The EVALUATE procedure updates the values $|c_2| \leftarrow 0$ and $d(c_2, x_8) \leftarrow 0$ (product $x_2x_8^2$ becomes inactive). As a result $|c_2| = 0$ is satisfied and so we set $p \leftarrow$ **false** and backtrack to level 2 (see row 4 in Tab. 1). There we change the assignment $x_8 \leftarrow 1$ and proceed.

Level 3, $i = 8$, $v = 1$. The EVALUATE procedure finds now that the product $x_2x_8 \in c_4$ is equal to one. So we set $p \leftarrow$ **false** and backtrack to level 2 again (see row 5 in Tab. 1). We unset variable x_8, backtrack to level 1 and change the value $x_6 \leftarrow 1$.

Level 2, $i = 6$, $v = 1$. The EVALUATE procedure does not find any contradictions nor c_2 satisifed, so we set $p \leftarrow$ **true** and continue (see row 6 in Tab. 1). In the REORDER procedure we find that there are two most frequent variables x_1 and x_8. In case of tie we pick the variable with lower index, so we pick x_1 and proceed.

Level 3, $i = 1$, $v = 0$. The EVALUATE procedure updates the values $|c_2|$ and $d(c_2, x_i)$ as follows: $|c_2| \leftarrow 2$, $d(c_2, x_1) \leftarrow 0$, $d(c_2, x_5) \leftarrow 0$, $d(c_2, x_8) \leftarrow 1$ (products $x_1x_5x_6$ and $x_1x_6x_8$ become inactive). We proceed with $p \leftarrow$ **true** as $|c_2| \neq 0$ (see row 7 in Tab. 1). In the REORDER procedure we choose variable x_7 (as $d(c_2, x_7) = d(c_2, x_8) = 1$ and $7 < 8$) and make another recursive call.

Level 4, $i = 7$, $v = 0$. The EVALUATE procedure updates the values $|c_2| \leftarrow 1$ and $d(c_2, x_7) \leftarrow 0$ (product $x_2x_6x_7$ becomes inactive), and proceed with $p \leftarrow$ **true** as $|c_2| \neq 0$ (see row 8 in Tab. 1). In the REORDER procedure we choose variable x_8 as the only one remaining and set it to zero.

Level 5, $i = 8$, $v = 0$. As before we find a contradiction in c_2, as the number of active products $|c_2| = 0$ and we backtrack (see row 9 in Tab. 1). Then we set $x_8 \leftarrow 1$.

Level 5, $i = 8$, $v = 1$. As before we find a contradiction in c_4, since the product $x_2x_8 = 1$ so we backtrack (see row 10 in Tab. 1), unset variable x_8 and change the value of $x_7 \leftarrow 1$.

Level 4, $i = 7$, $v = 1$. We find no contradiction in constraints c_3–c_5, but we find that the product $x_2x_6x_7 \in c_2$ is equal to one, and so constraint c_2 is satisfied. Since both c_1 and c_2 are satisfied we set $f \leftarrow$ **true** and terminate the procedure (see row 11 in Tab. 1).

The final results of the execution of the INDUCENFA procedure using the *min-max-ex* variable ordering method are gathered in Tab. 1. The resulting automaton is shown in Fig. 1.

Table 1: INDUCENFA algorithm execution for the *min-max-ex* variable ordering method and $y_1 = 0$, $y_2 = 1$. No. – step number, c_+ – selected constraint, x_i, v – next variable and value to set, Result – result of the EVALUATE procedure.

No.	c_+	x_i	v	Result
1.	c_1	x_2	0	$p \leftarrow$ **false** due to c_1
2.	-	x_2	1	c_1 satisfied
3.	c_2	x_6	0	$p \leftarrow$ **true**
4.	c_2	x_8	0	$p \leftarrow$ **false** due to c_2
5.	-	x_8	1	$p \leftarrow$ **false** due to c_4
6.	-	x_6	1	$p \leftarrow$ **true**
7.	c_2	x_1	0	$p \leftarrow$ **true**
8.	c_2	x_7	0	$p \leftarrow$ **true**
9.	c_2	x_8	0	$p \leftarrow$ **false** due to c_2
10.	-	x_8	1	$p \leftarrow$ **false** due to c_4
11.	-	x_7	1	c_2 satisfied, $x_1 \leftarrow 0$, $x_3 \leftarrow 0$, $x_4 \leftarrow 0$, $x_5 \leftarrow 0$, $x_8 \leftarrow 0$, $f \leftarrow$ **true**

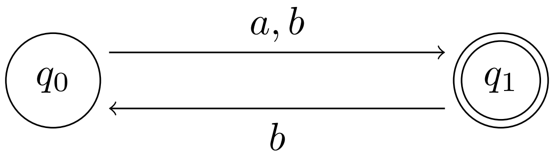

Figure 1: An NFA consistent with the sample $S = (\{a, abb\}, \{aaa, ab, ba\})$

2 The *min-max-cex* variable ordering method

Level 0, $i = \emptyset$, $v = \emptyset$. At first the REORDER procedure is called. The constraints related to counterexamples having the fewest active products are c_4 and c_5, but we choose the one with the lower index, i.e., c_4. The constraint contains four equally frequent variables, and we choose x_1 as the next variable. Therefore, $i \leftarrow 1$ and $v \leftarrow 1$, as the REORDER procedure now always assigs one first.

Level 1, $i = 1$, $v = 1$. Since $v = 1$, the EVALUATE procedure checks whether any of the constraints c_3–c_5 contains a product equal to one. Since there is no such product we proceed with $p \leftarrow$ **true** (see row 1 in Tab. 2). In the REORDER procedure we again choose constraint c_4 as the one with the fewest products, and we choose x_2 as the next variable.

Level 2, $i = 2$, $v = 1$. The EVALUATE procedure now checks whether any of the constraints c_3–c_5 contains a product equal to one. As the product $x_1^2x_2 \in c_3$ becomes equal to one we backtrack and change the value of x_2 (see row 2 in Tab. 2).

Level 2, $i = 2$, $v = 0$. The EVALUATE procedure checks now that with $x_2 = 0$ the number of active products in c_1 becomes zero, so we need to backtrack once again. We unset the variable x_2 and change the value of $x_1 \leftarrow 0$ (see row 3 in Tab. 2).

Level 1, $i = 1$, $v = 0$. The EVALUATE procedure does not find any contradiction in either c_1 or c_2. Hence, it updates the values $|c_3| \leftarrow 2$, $|c_4| \leftarrow 1$, $d(c_3, x_1) \leftarrow 0$, $d(c_3, x_2) \leftarrow 2$, $d(c_3, x_4) \leftarrow 1$, and $d(c_4, x_6) \leftarrow 0$ (products $x_1^2x_2$, $x_1x_2x_4$, and x_1x_6 become inactive). We set $p \leftarrow$ **true** (as none of the constraints c_3–c_5 became satisfied) and continue (see row 4 in Tab. 2). The REORDER procedure chooses constraint c_4 and variable x_2.

Level 2, $i = 2$, $v = 1$. The EVALUATE procedure finds no contradiction in c_3–c_5, so we proceed with $p \leftarrow$ **true** (see row 5 in Tab. 2). In the REORDER procedure we choose constraint c_4 again and variable x_8, the only remaining variable of this constraint.

Level 3, $i = 8$, $v = 1$. The EVALUATE procedure finds that the product $x_2x_8 \in c_4$ is equal to one, so we return $p \leftarrow$ **false** and backtrack (see row 6 in Tab. 2).

Level 3, $i = 8$, $v = 0$. The EVALUATE procedure finds no contradiction in c_1 and c_2 and updates the values $|c_4| \leftarrow 0$ and $d(c_4, x_8) \leftarrow 0$ (product x_2x_8 becomes inactive). This way c_4 becomes satisfied (see row 7 in Tab. 2) and we continue. In the REORDER procedure we choose constraint c_3 (as $|c_3| = |c_5|$, but $3 < 5$) and we set $x_3 \leftarrow 1$.

Level 4, $i = 3$, $v = 1$. Setting $x_3 \leftarrow 1$ makes the EVALUATE procedure detect a contradiction in constraint c_3, which sets $p \leftarrow$ **false** and forces backtracking (see row 8 in Tab. 2).

Level 4, $i = 3$, $v = 0$. As the current setting does not affect constraints c_1 and c_2, we update the values $|c_3| \leftarrow 1$ and $d(c_3, x_3) \leftarrow 0$ (product $x_2^2x_3$ becomes inactive). We set $p \leftarrow$ **true** (see row 9 in Tab. 2) and call REORDER procedure. In this procedure we choose c_3 and variable x_4.

Level 5, $i = 4$, $v = 1$. Setting $x_4 \leftarrow 1$ makes the EVALUATE procedure detect a contradiction in constraint c_3, which sets $p \leftarrow$ **false** and forces backtracking (see row 10 in Tab. 2).

Level 5, $i = 4$, $v = 0$. Variable x_4 does not appear in constraints c_1 and c_2, so we set $|c_3| \leftarrow 0$, $|c_5| \leftarrow 1$, and $d(c_3, x_4) \leftarrow 0$ and we mark constraint c_3 as satisfied (see row 11 in Tab. 2). In the REORDER procedure we select constraint c_5 as the only not-yet-satisfied constraint related to counterexamples and we pick variable x_5.

Level 6, $i = 5$, $v = 1$. The assignment $x_5 \leftarrow 1$ causes a contradiction in c_5 so we set $p \leftarrow$ **false** and backtrack (see row 12 in Tab. 2).

Level 6, $i = 5$, $v = 0$. The assignment $x_5 \leftarrow 0$ causes the constraint c_5 to become satisifed (again c_1 and c_2 are not affected). Since all constraints c_3–c_5 are satisfied, we set $x_6 \leftarrow 1$, $x_7 \leftarrow 1$ and $f \leftarrow$ **true** and terminate the procedure (see row 13 in Tab. 2).

The final results of the execution of the INDUCENFA procedure using the *min-max-cex* variable ordering method are gathered in Tab. 2. The resulting automaton is the same as the NFA in Fig. 1.

3 The *deg* variable ordering method

When using the *deg* method we first establish the order or variables based on the value $D(x_j) = \sum_{i=1}^{5} d(c_i, x_j)$, for $j \in [1, 8]$. These values are as follows: $D(x_1) = 5$, $D(x_2) = 9$, $D(x_3) = 1$, $D(x_4) = 3$, $D(x_5) = 2$, $D(x_6) = 5$, $D(x_7) = 1$, $D(x_8) = 3$. Sorting by $D(x_j)$ and in case of ties picking the variable with smaller index, we get the following order of variables: x_2, x_1, x_6, x_4, x_8, x_5, x_3, and x_7.

Table 2: InduceNFA algorithm execution for *min-max-cex* variable ordering method and $y_1 = 0$, $y_2 = 1$. No. – step number, c_- – selected constraint, x_i, v – next variable and value to set, Result – result of Evaluate procedure.

No.	c_-	x_i	v	Result
1.	c_4	x_1	1	$p \leftarrow$ **true**
2.	c_4	x_2	1	$p \leftarrow$ **false** due to c_3
3.	-	x_2	0	$p \leftarrow$ **false** due to c_1
4.	-	x_1	0	$p \leftarrow$ **true**
5.	c_4	x_2	1	$p \leftarrow$ **true**
6.	c_4	x_8	1	$p \leftarrow$ **false** due to c_4
7.	-	x_8	0	c_4 satisfied
8.	c_3	x_3	1	$p \leftarrow$ **false** due to c_3
9.	-	x_3	0	$\lvert c_3\rvert \leftarrow 1$
10.	c_3	x_4	1	$p \leftarrow$ **false** due to c_3
11.	-	x_4	0	c_3 satisfied, $\lvert c_5\rvert \leftarrow 1$
12.	c_5	x_5	1	$p \leftarrow$ **false** due to c_5
13.	-	x_5	0	c_5 satisifed, $x_6 \leftarrow 1$, $x_7 \leftarrow 1$, $f \leftarrow$ **true**

Level 0, $i = \emptyset$, $v = \emptyset$. The Reorder procedure simply return x_2 with the initial assignment $x_2 \leftarrow 0$.

Level 1, $i = 2$, $v = 0$. Since $v = 0$, the Evaluate procedure checks first whether there is any contradiction in the constraints c_1 and c_2. Since $|c_1| = 0$ holds, we need to backtrack (see row 1 in Tab. 3).

Level 1, $i = 2$, $v = 1$. The Evaluate procedure now checks whether any of the constraints c_3–c_5 contains a product equal to one. As no such product is found, we check whether contraint c_1 or c_2 is satisfied. We mark constraint c_1 as satisifed and proceed (see row 2 in Tab. 3). The Reorder procedure returns variable x_1.

Level 2, $i = 1$, $v = 0$. The Evaluate procedure checks that there is no contradiction with respect to c_2 and updates the values $|c|$ and $d(c, x_i)$ in the following way: $|c_2| \leftarrow 2$, $|c_3| \leftarrow 2$, $|c_4| \leftarrow 1$, $d(c_2, x_1) \leftarrow 0$, $d(c_2, x_5) \leftarrow 0$, $d(c_2, x_6) \leftarrow 1$, $d(c_2, x_8) \leftarrow 1$, $d(c_3, x_1) \leftarrow 0$, $d(c_3, x_4) \leftarrow 1$, $d(c_4, x_1) \leftarrow 0$ and $d(c_4, x_6) \leftarrow 0$ (products $x_1x_5x_6$, $x_1x_6x_8$, $x_1^2x_2$, $x_1x_2x_4$, and x_1x_6 become inactive). We proceed with $p \leftarrow$ **true** (see row 3 in Tab. 3). The Reorder procedure returns variable x_6.

Level 3, $i = 6$, $v = 0$. As there is no contrdiction, the Evaluate procedure updates the values $|c_2| \leftarrow 1$, $|c_5| \leftarrow 1$, $d(c_2, x_6) \leftarrow 0$, $d(c_2, x_7) \leftarrow 0$, and $d(c_5, x_4) \leftarrow 0$ (products x_2x_6x7 and x_4x_6 become inactive). We set $p \leftarrow$ **true** and continue (see row 4 in Tab. 3). The Reorder procedure chooses returns variable x_4.

Level 4, $i = 4$, $v = 0$. Since x_4 does not appear in c_2, the Evaluate procedure updates the values $|c_3| \leftarrow 1$ and $d(c_3, x_4) \leftarrow 0$ (product $x_2x_4^2$ becomes inactive). We set $p \leftarrow$ **true** and continue (see row 5 in Tab. 3). The Reorder procedure returns variable x_8.

Level 5, $i = 8$, $v = 0$. Setting $x_8 \leftarrow 0$ causes a contradiction in c_2, so we need to backtrack (see row 6 in Tab. 3).

Level 5, $i = 8$, $v = 1$. Setting $x_8 \leftarrow 1$ causes a contradiction in c_4, so we need to backtrack again (see row 7 in Tab. 3). We unset variable x_8 and change the value $x_4 \leftarrow 1$.

Level 4, $i = 4$, $v = 1$. Setting $x_4 \leftarrow 1$ causes a contradiction in c_3, so we need to backtrack again (see row 8 in Tab. 3). We unset variable x_4 and change the value $x_6 \leftarrow 1$.

Level 3, $i = 6$, $v = 1$. The EVALUATE procedure does not detect any contradiction, so we proceed with $p \leftarrow$ **true** (see row 9 in Tab. 3). The REORDER procedure returns variable x_4.

Level 4, $i = 4$, $v = 0$. The EVALUATE procedure does not detect any contradiction, so we proceed with $p \leftarrow$ **true** (see row 10 in Tab. 3). The REORDER procedure returns variable x_8.

Level 5, $i = 8$, $v = 0$. There is no contradiction in c_2, but the EVALUATE procedure detects that c_4 is satisfied. So we continue with $p \leftarrow$ **true** (see row 11 in Tab. 3). The REORDER procedure returns variable x_5.

Level 6, $i = 5$, $v = 0$. The assignment $x_5 \leftarrow 0$ makes constraint c_5 satisfied (product x_2x_5 becomes inactive, and $|c_5| \leftarrow 0$), so we set $p \leftarrow$ **true** (see row 12 in Tab. 3). The REORDER procedure returns variable x_3.

Level 7, $i = 3$, $v = 0$. The assignment $x_3 \leftarrow 0$ makes constraint c_3 satisifed (product $x_2^2x_3$ becomes inactive and $|c_3| \leftarrow 0$), so we continue (see row 13 in Tab. 3). The REORDER procedure returns variable x_7.

Level 8, $i = 7$, $v = 0$. The assignment $x_7 \leftarrow 0$ causes a contradiction in constraint c_2 (as product $x_2x_6x_7$ becomes inactive and so $|c_2| \leftarrow 0$). We set $p \leftarrow$ **false** and backtrack (see row 14 in Tab. 3).

Level 8, $i = 7$, $v = 1$. All the constraints become satisifed as the product $x_2x_6x_7 \in c_2$ becomes equal to one. We set $f \leftarrow$ **true** and terminate the procedure without setting any other variable (see row 15 in Tab. 3).

The final results of the execution of the INDUCENFA procedure using the *deg* variable ordering method are gathered in Tab. 3. The resulting automaton is the same as the NFA in Fig. 1.

Table 3: INDUCENFA algorithm execution for *deg* variable ordering method and $y_1 = 0$, $y_2 = 1$. No. – step number, c_- – selected constraint, x_i, v – next variable and value to set, Result – result of EVALUATE procedure.

No.	c_-	x_i	v	Result
1.	c_4	x_1	1	$p \leftarrow$ **true**
2.	c_4	x_2	1	$p \leftarrow$ **false** due to c_3
3.	-	x_2	0	$p \leftarrow$ **false** due to c_1
4.	-	x_1	0	$p \leftarrow$ **true**
5.	c_4	x_2	1	$p \leftarrow$ **true**
6.	c_4	x_8	1	$p \leftarrow$ **false** due to c_4
7.	-	x_8	0	c_4 satisfied
8.	c_3	x_3	1	$p \leftarrow$ **false** due to c_3
9.	-	x_3	0	$\lvert c_3 \rvert \leftarrow 1$
10.	c_3	x_4	1	$p \leftarrow$ **false** due to c_3
11.	-	x_4	0	c_3 satisfied, $\lvert c_5 \rvert \leftarrow 1$
12.	c_5	x_5	1	$p \leftarrow$ **false** due to c_5
13.	-	x_5	0	c_5 satisifed, $x_6 \leftarrow 1$, $x_7 \leftarrow 1$, $f \leftarrow$ **true**
14.	-	x_5	0	c_5 satisifed, $x_6 \leftarrow 1$, $x_7 \leftarrow 1$, $f \leftarrow$ **true**
15.	-	x_5	0	c_5 satisifed, $x_6 \leftarrow 1$, $x_7 \leftarrow 1$, $f \leftarrow$ **true**