Algorithm 1 CheckSafe

```
Input: The program, an SCC, the entry transitions of the SCC, an exit transition with an assertion
if the exit transition already implies the assertion then
    return Safe
else if the exit transition is an initial transition then
    return Maybe
Call CondSafe for the SCC with the given assertion
if no CII could be found then
    return Maybe
for all entry SCCs do
    for all literals of the according condition of the CII do
        Call CheckSafe for the entry SCC and with the literal as assertion
if all calls returned Safe then
    return Safe
return the result of a call to CheckSafe with a narrowed version
```

Algorithm 2 CondSafe

```
Input: An SCC, the entry transitions of the SCC, an exit transition with an assertion
$k \leftarrow 1$
**repeat**
    Construct a formula $\mathbb{F}_k$ for the SCC and the assertion
    Call the Max-SMT-Solver with $\mathbb{F}_k$
    **if** it returned a solution **then**
        **return** an invariant assigning each location the according condition from $\mathbb{F}_k$
    $k \leftarrow k + 1$
**until** $k >$ MAX_CONJUNCTS
**return** None
```

$$I_{\ell,k}(\mathcal{V}) \equiv \bigwedge_{1\leq j\leq k} I_{\ell,j,k}(\mathcal{V}) \tag{1}$$

$$I_{\ell,j,k}(\mathcal{V}) \equiv i_{\ell,j} + \sum_{v\in\mathcal{V}} i_{\ell,j,v} * v \leq 0 \tag{2}$$

$$\mathbb{I}_{t,j,k} \equiv \tau \Rightarrow I'_{\ell',j,k} \tag{3}$$

$$\mathbb{C}_{t,k} \equiv I_{\ell,k} \wedge \tau \Rightarrow I'_{\ell',k} \tag{4}$$

$$\mathbb{S}_{k} \equiv I_{\tilde{\ell}_{\text{exit}},k} \wedge \tau_{\text{exit}} \Rightarrow \varphi' \tag{5}$$

$$\mathbb{F}_{k} \equiv \bigwedge_{t\in\mathcal{C}} \mathbb{C}_{t,k} \wedge \mathbb{S}_{k} \wedge \bigwedge_{t\in\mathcal{E}_{\mathcal{C}},1\leq j\leq k} (\mathbb{I}_{t,j,k} \vee \neg p_{\mathbb{I}_{t,j,k}}) \wedge \bigwedge_{t\in\mathcal{E}_{\mathcal{C}},1\leq j\leq k} [p_{\mathbb{I}_{t,j,k}}, \omega_{\mathbb{I}}] \tag{6}$$

Algorithm 3 Narrowing

```
for all entry transitions do
    for all literals of the CII do
        if literal could not be proved safe for this transition then
            Add a conjunct with the negated literal to the transition
for all transitions of the SCC do
    Add a conjunct with the negated CII at the start location to the transition
    Add a conjunct with the negated CII at the end location to the transition
```