Non-deterministic Characterisations

Cynthia Kop

5 September, 2016

Overview

Overview

Complexity Classes

Complexity Classes

- classes like P

Complexity Classes

- classes like P, NP

Complexity Classes

- classes like P, NP, EXP

Complexity Classes

- classes like P, NP, EXP, LOGSPACE

Complexity Classes

- classes like P, NP, EXP, LOGSPACE, ...

Complexity Classes

- classes like P, NP, EXP, LOGSPACE, ...

decision functions which can be evaluated in polynomial time

Characterising Classes

Suppose:

Characterising Classes

Suppose:

- every function in P can be implemented by (e.g.) a functional program with properties X, Y, Z

Characterising Classes

Suppose:

- every function in P can be implemented by (e.g.) a functional program with properties X, Y, Z
- every function implemented by a functional program with properties X, Y, Z is in P

Characterising Classes

Suppose:

- every function in P can be implemented by (e.g.) a functional program with properties X, Y, Z
- the outcome of every functional program with properties X, Y, Z can be evaluated by a deterministic Turing Machine in polynomial time

Characterising Classes

Suppose:

- every function in P can be implemented by (e.g.) a functional program with properties X, Y, Z
- the outcome of every functional program with properties X, Y, Z can be evaluated by a deterministic Turing Machine in polynomial time
- $\Longrightarrow$ then F in P if and only if F can be implemented as a functional program with properties X, Y, Z

Characterising Classes

Suppose:

- every function in P can be implemented by (e.g.) a functional program with properties X, Y, Z
- the outcome of every functional program with properties X, Y, Z can be evaluated by a deterministic Turing Machine in polynomial time
- $\Longrightarrow$ then F in P if and only if F can be implemented as a functional program with properties X, Y, Z
- $\Longrightarrow$ "functional programs with properties X, Y, Z" characterise P

Characterising Classes

Suppose:

- every function in P can be implemented by (e.g.) a functional program with properties X, Y, Z
- the outcome of every functional program with properties X, Y, Z can be evaluated by a deterministic Turing Machine in polynomial time
- $\Longrightarrow$ then F in P if and only if F can be implemented as a functional program with properties X, Y, Z
- $\Longrightarrow$ "functional programs with properties X, Y, Z" characterise P

Characterising Classes

Suppose:

- every function in NP can be implemented by (e.g.) a functional program with properties X, Y, Z
- the outcome of every functional program with properties X, Y, Z can be evaluated by a non-deterministic Turing Machine in polynomial time
- $\Longrightarrow$ then F in NP if and only if F can be implemented as a functional program with properties X, Y, Z
- $\Longrightarrow$ "functional programs with properties X, Y, Z" characterise NP

Characterising Classes

Suppose:

- every function in NP can be implemented by (e.g.) a functional program with properties X, Y, Z
- the outcome of every functional program with properties X, Y, Z can be evaluated by a non-deterministic Turing Machine in polynomial time
- $\Longrightarrow$ then F in NP if and only if F can be implemented as a functional program with properties X, Y, Z
- $\Longrightarrow$ "functional programs with properties X, Y, Z" characterise NP

Result: question "$\texttt{P} = \texttt{NP}$?" reduced to: "does property Z limit expressivity (in context)?"

In 2001, Jones:

In 2001, Jones:

- defined a minimal functional programming language operating on lists

In 2001, Jones:

- defined a minimal functional programming language operating on lists
- introduced *cons-free* programs, which cannot generate new data

In 2001, Jones:

- defined a minimal functional programming language operating on lists
- introduced *cons-free* programs, which cannot generate new data
- showed that programs of data order k can simulate arbitrary deterministic Turing Machines in $\mathcal{O}(\exp_2^k(n))$

In 2001, Jones:

- defined a minimal functional programming language operating on lists
- introduced *cons-free* programs, which cannot generate new data
- showed that programs of data order k can simulate arbitrary deterministic Turing Machines in $\mathcal{O}(\mathtt{exp}_2^k(n))$
- showed that arbitrary programs of data order k can be simulated by deterministic Turing Machines in $\mathcal{O}(\mathtt{exp}_2^k(n))$

In 2001, Jones:

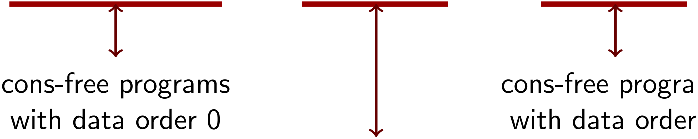
- defined a minimal functional programming language operating on lists
- introduced *cons-free* programs, which cannot generate new data
- showed that programs of data order k can simulate arbitrary deterministic Turing Machines in $\mathcal{O}(\mathrm{exp}_2^k(n))$
- showed that arbitrary programs of data order k can be simulated by deterministic Turing Machines in $\mathcal{O}(\mathrm{exp}_2^k(n))$
- $\Longrightarrow$ proved that programs of data order k *characterise* EXPTIME^k

In 2001, Jones:

- defined a minimal functional programming language operating on lists
- introduced *cons-free* programs, which cannot generate new data
- showed that programs of data order k can simulate arbitrary deterministic Turing Machines in $\mathcal{O}(\mathtt{exp}_2^k(n))$
- showed that arbitrary programs of data order k can be simulated by deterministic Turing Machines in $\mathcal{O}(\mathtt{exp}_2^k(n))$
- $\Longrightarrow$ proved that programs of data order k *characterise* EXPTIME^k
- (proved several other characterisations as well)

$$\mathsf{P} = \mathsf{EXPTIME}^0 \quad \subset \quad \mathsf{EXPTIME}^1 \quad \subset \quad \mathsf{EXPTIME}^2 \quad \subseteq \quad \dots$$

$$\underline{\mathsf{P} = \mathsf{EXPTIME}^0} \subset \underline{\mathsf{EXPTIME}^1} \subset \underline{\mathsf{EXPTIME}^2} \subseteq \ldots$$



cons-free programs with data order 0

cons-free programs with data order 1

cons-free programs with data order 2

In 2001, Jones

- showed that cons-free programs with data order k characterise EXPTIME^k

In 2001, Jones

- showed that deterministic cons-free programs with data order k characterise $\mathsf{EXPTIME}^k$

In 2001, Jones

- showed that deterministic cons-free programs with data order k characterise EXPTIME^k

In 2006, Bonfante

- showed that non-deterministic cons-free programs with data order 0 ...

In 2001, Jones

- showed that deterministic cons-free programs with data order k characterise $\mathsf{EXPTIME}^k$

In 2006, Bonfante

- showed that non-deterministic cons-free programs with data order 0 characterise $\mathsf{P} = \mathsf{EXPTIME}^0$

In 2001, Jones

- showed that deterministic cons-free programs with data order k characterise EXPTIME^k

In 2006, Bonfante

- showed that non-deterministic cons-free programs with data order 0 characterise P $=$ EXPTIME^0

In 2016, we

In 2001, Jones

- showed that deterministic cons-free programs with data order k characterise $\mathsf{EXPTIME}^k$

In 2006, Bonfante

- showed that non-deterministic cons-free programs with data order 0 characterise $\mathsf{P} = \mathsf{EXPTIME}^0$

In 2016, we

- wondered whether non-deterministic cons-free programs with data order k characterise $\mathsf{EXPTIME}^k$

In 2001, Jones

- showed that deterministic cons-free programs with data order k characterise EXPTIME^k

In 2006, Bonfante

- showed that non-deterministic cons-free programs with data order 0 characterise P $=$ EXPTIME^0

In 2016, we

- wondered whether non-deterministic cons-free programs with data order k characterise EXPTIME^k
- found out that they don't

$\underline{\mathsf{P} = \mathsf{EXPTIME}^0} \subset \underline{\mathsf{EXPTIME}^1} \subset \underline{\mathsf{EXPTIME}^2} \subseteq \dots$

$\mathsf{P} = \mathsf{EXPTIME}^0$ ↕ cons-free programs with data order 0

$\mathsf{EXPTIME}^1$ ↕ cons-free programs with data order 1

$\mathsf{EXPTIME}^2$ ↕ cons-free programs with data order 2

$$\mathsf{P} = \mathsf{EXPTIME}^0 \quad \subset \quad \mathsf{ELEMENTARY} = \bigcup_{k \in \mathbb{N}} \mathsf{EXPTIME}^k$$

cons-free non-deterministic programs with data order 0

cons-free non-deterministic programs with data order $k > 0$

Overview

Basics

Basics

- standard functional programming language:

Basics

- standard functional programming language: applicative term rewriting system with simple types and constructor rules

Basics

- standard functional programming language: applicative term rewriting system with simple types and constructor rules
- pair types are allowed

Basics

- standard functional programming language: applicative term rewriting system with simple types and constructor rules
- pair types are allowed
- constructors must be fully applied, and have a data type as output type

Basics

- standard functional programming language: applicative term rewriting system with simple types and constructor rules
- pair types are allowed
- constructors must be fully applied, and have a data type as output type
- call-by-value reduction

Basics

- standard functional programming language: applicative term rewriting system with simple types and constructor rules
- pair types are allowed
- constructors must be fully applied, and have a data type as output type
- call-by-value reduction
- values are:
  - ground terms built from constructors (data)
  - incomplete function applications $f\ v_1 \cdots v_n$ with all v_i values

Cons-free programs

Cons-free programs

- in Jones: programs not containing the list constructor ("cons-free")

Cons-free programs

- in Jones: programs not containing the list constructor ("cons-free")
- in practice: no new data is created

Cons-free programs

- in Jones: programs not containing the list constructor ("cons-free")
- in practice: no new data is created
- in rules $\ell \to r$, the right-hand side r:

Cons-free programs

- in Jones: programs not containing the list constructor ("cons-free")
- in practice: no new data is created
- in rules $\ell \to r$, the right-hand side r:
  - *may* have subterms which are data

Cons-free programs

- in Jones: programs not containing the list constructor ("cons-free")
- in practice: no new data is created
- in rules $\ell \to r$, the right-hand side r:
  - *may* have subterms which are data (e.g. true, s 0, but not s alone)

Cons-free programs

- in Jones: programs not containing the list constructor ("cons-free")
- in practice: no new data is created
- in rules $\ell \to r$, the right-hand side r:
  - *may* have subterms which are data (e.g. true, s 0, but not s alone)
  - *may* have subterms occurring in ℓ

Cons-free programs

- in Jones: programs not containing the list constructor ("cons-free")
- in practice: no new data is created
- in rules $\ell \to r$, the right-hand side r:
  - *may* have subterms which are data (e.g. true, s 0, but not s alone)
  - *may* have subterms occurring in ℓ
  - *may not* contain constructors otherwise

Cons-free Higher-order Term Rewriting:

- right-hand sides:
  - *may* have subterms which are data
  - *may* have subterms which are also subterms of the left
  - *may not* contain constructors otherwise

Cons-free Higher-order Term Rewriting:

- right-hand sides:
  - *may* have subterms which are data
  - *may* have subterms which are also subterms of the left
  - *may not* contain constructors otherwise

$$\begin{array}{rcl} \mathsf{Hd}\ (h{::}t) & \to & h \\ \mathsf{Tl}\ (h{::}t) & \to & t \end{array}$$

Cons-free Higher-order Term Rewriting:

- right-hand sides:
  - *may* have subterms which are data
  - *may* have subterms which are also subterms of the left
  - *may not* contain constructors otherwise

$$
\begin{array}{rcl}
\mathsf{Evenlength}\ l & \rightarrow & \mathsf{Helper}\ l\ \mathsf{True} \\
\mathsf{Helper}\ []\ b & \rightarrow & b \\
\mathsf{Helper}\ (h{::}t)\ \mathsf{True} & \rightarrow & \mathsf{Helper}\ t\ \mathsf{False} \\
\mathsf{Helper}\ (h{::}t)\ \mathsf{False} & \rightarrow & \mathsf{Helper}\ t\ \mathsf{True}
\end{array}
$$

Cons-free Higher-order Term Rewriting:

- right-hand sides:
  - *may* have subterms which are data
  - *may* have subterms which are also subterms of the left
  - *may not* contain constructors otherwise

$$\begin{array}{rcl} \mathsf{Half}\ \mathsf{0} & \to & \mathsf{0} \\ \mathsf{Half}\ (\mathsf{S}\ \mathsf{0}) & \to & \mathsf{0} \\ \mathsf{Half}\ (\mathsf{S}\ (\mathsf{S}(x))) & \to & \mathsf{S}\ (\mathsf{Half}\ x) \end{array}$$

Cons-free Higher-order Term Rewriting:

- right-hand sides:
  - *may* have subterms which are data
  - *may* have subterms which are also subterms of the left
  - *may not* contain constructors otherwise

$$\begin{array}{rcl} \mathsf{Half}\ \mathsf{0} & \to & \mathsf{0} \\ \mathsf{Half}\ (\mathsf{S}\ \mathsf{0}) & \to & \mathsf{0} \\ \mathsf{Half}\ (\mathsf{S}\ (\mathsf{S}(x))) & \to & \mathsf{S}\ (\mathsf{Half}\ x) \end{array}$$

Cons-free Higher-order Term Rewriting:

- right-hand sides:
  - *may* have subterms which are data
  - *may* have subterms which are also subterms of the left
  - *may not* contain constructors otherwise

$$
\begin{array}{rcl}
\mathsf{Half}\ x & \to & \mathsf{Helper}\ x\ x \\
\mathsf{Helper}\ \mathsf{0}\ y & \to & y \\
\mathsf{Helper}\ (\mathsf{S}\ \mathsf{0})\ (\mathsf{S}\ y) & \to & y \\
\mathsf{Helper}\ (\mathsf{S}\ (\mathsf{S}\ x))\ (\mathsf{S}\ y) & \to & \mathsf{Helper}\ x\ y
\end{array}
$$

Cons-free Higher-order Term Rewriting:

- right-hand sides:
  - *may* have subterms which are data
  - *may* have subterms which are also subterms of the left
  - *may not* contain constructors otherwise

$$\begin{array}{rcl} \mathsf{Last}\ (h{::}\,[\,]) & \to & h \\ \mathsf{Last}\ (h_1{::}h_2{::}tl) & \to & \mathsf{Last}\ (h_2{::}tl) \end{array}$$

Cons-free Higher-order Term Rewriting:

- right-hand sides:
  - *may* have subterms which are data
  - *may* have subterms which are also subterms of the left
  - *may not* contain constructors otherwise

$$\begin{array}{rcl} \mathsf{Last}\ (h{::}\,[\,]) & \to & h \\ \mathsf{Last}\ (h_1{::}h_2{::}tl) & \to & \mathsf{Last}\ (h_2{::}tl) \end{array}$$

Cons-free Higher-order Term Rewriting:

- right-hand sides:
  - *may* have subterms which are data
  - *may* have subterms which are also subterms of the left
  - *may not* contain constructors otherwise

$$\begin{array}{rcl} \mathsf{fold}\ F\ z\ [] & \to & z \\ \mathsf{fold}\ F\ z\ (h{::}tl) & \to & F\ h\ (\mathsf{fold}\ F\ z\ tl) \end{array}$$

Cons-free Higher-order Term Rewriting:

- right-hand sides:
  - *may* have subterms which are data
  - *may* have subterms which are also subterms of the left
  - *may not* contain constructors otherwise

$$\begin{array}{rcl} \mathsf{map}\ F\ [] & \to & [] \\ \mathsf{map}\ F\ (h{::}tl) & \to & (F\ h){::}(\mathsf{map}\ F\ tl) \end{array}$$

Cons-free Higher-order Term Rewriting:

- right-hand sides:
  - *may* have subterms which are data
  - *may* have subterms which are also subterms of the left
  - *may not* contain constructors otherwise

$$\begin{array}{rcl} \mathsf{map}\ F\ [] & \to & [] \\ \mathsf{map}\ F\ (h{::}tl) & \to & (F\ h){::}(\mathsf{map}\ F\ tl) \end{array}$$

Overview

Recall:

$$\mathsf{P} = \mathsf{EXPTIME}^0 \quad \subset \quad \mathsf{ELEMENTARY} = \bigcup_{k \in \mathbb{N}} \mathsf{EXPTIME}^k$$

P = EXPTIME^0 ↔ cons-free non-deterministic programs with data order 0

ELEMENTARY ↔ cons-free non-deterministic programs with data order $k > 0$

We show:

We show:

- the decision problem accepted by a non-deterministic cons-free program with data order 0 (so a first-order cons-free TRS) is in P

We show:

- the decision problem accepted by a non-deterministic cons-free program with data order 0 (so a first-order cons-free TRS) is in P
- a non-deterministic cons-free program with data order 0 can decide any decision problem in P

We show:

- the decision problem accepted by a non-deterministic cons-free program with data order 0 (so a first-order cons-free TRS) is in P
- a non-deterministic cons-free program with data order 0 can decide any decision problem in P
- the decision problem accepted by a non-deterministic cons-free program is in ELEMENTARY regardless of data order

We show:

- the decision problem accepted by a non-deterministic cons-free program with data order 0 (so a first-order cons-free TRS) is in P
- a non-deterministic cons-free program with data order 0 can decide any decision problem in P
- the decision problem accepted by a non-deterministic cons-free program is in ELEMENTARY regardless of data order
- a non-deterministic cons-free program with data order 1 can decide any decision problem in ELEMENTARY

We show:

- the decision problem accepted by a non-deterministic cons-free program with data order 0 (so a first-order cons-free TRS) is in P
- a non-deterministic cons-free program with data order 0 can decide any decision problem in P
- the decision problem accepted by a non-deterministic cons-free program is in ELEMENTARY regardless of data order
- a non-deterministic cons-free program with data order 1 can decide any decision problem in ELEMENTARY

Constructing an algorithm

Constructing an algorithm

Rules:

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Constructing an algorithm

Rules:

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s,\mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Start term: $\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[])$

Constructing an algorithm

Rules:

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s,\mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Start term: $\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[])$

Constructing an algorithm

Rules:

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Start term: $\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[])$

Let $\mathcal{B} =$

Constructing an algorithm

Rules:

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \\ \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Start term: $\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[])$

Let $\mathcal{B} =$

$\{0;0;1;[]$

Constructing an algorithm

Rules:

$$\begin{array}{rcl}
\mathsf{Subseq}([],\ t) & \rightarrow & \mathsf{True} \\
\mathsf{Subseq}(s,\ []) & \rightarrow & \mathsf{False} \\
\mathsf{tl}(x;y) & \rightarrow & y
\end{array}
\qquad
\begin{array}{rcl}
\mathsf{Subseq}(s,\ t) & \rightarrow & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(0;x,\ 0;t) & \rightarrow & \mathsf{Subseq}(s,\ t) \\
\mathsf{Subseq}(1;x,\ 1;t) & \rightarrow & \mathsf{Subseq}(s,\ t)
\end{array}$$

Start term: $\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[])$

Let $\mathcal{B} =$

$\{0;0;1;[],\ 0;1;[]$

Constructing an algorithm

Rules:

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Start term: $\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[])$

Let $\mathcal{B} =$

$\{0;0;1;[],\ 0;1;[],\ 1;[]$

Constructing an algorithm

Rules:

$$
\begin{array}{rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\
\mathsf{tl}(x;y) & \to & y
\end{array}
\qquad
\begin{array}{rcl}
\mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Start term: $\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[])$

Let $\mathcal{B} =$

$\{0;0;1;[],\ 0;1;[],\ 1;[],\ []$

Constructing an algorithm

Rules:

$$\begin{array}{rcl@{\qquad}rcl}
\mathsf{Subseq}([\,],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ [\,]) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}$$

Start term: $\mathsf{Subseq}(0;0;1;[\,],\ 0;1;0;1;[\,])$

Let $\mathcal{B} =$

$\{0;0;1;[\,],\ 0;1;[\,],\ 1;[\,],\ [\,],\ 0;1;0;1;[\,]$

Constructing an algorithm

Rules:

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Start term: $\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[])$

Let $\mathcal{B} =$

$\{0;0;1;[],\ 0;1;[],\ 1;[],\ [],\ 0;1;0;1;[],\ 1;0;1;[]$

Constructing an algorithm

Rules:

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Start term: $\mathsf{Subseq}(0;0;1;[],\ 0;1;\color{red}{0;1;[]})$

Let $\mathcal{B} =$

$\{0;0;1;[],\ \color{red}{0;1;[]},\ 1;[],\ [],\ 0;1;0;1;[],\ 1;0;1;[]$

Constructing an algorithm

Rules:

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s,\mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Start term: $\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[])$

Let $\mathcal{B} =$

$\{0;0;1;[],\ 0;1;[],\ 1;[],\ [],\ 0;1;0;1;[],\ 1;0;1;[]$

Constructing an algorithm

Rules:

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Start term: $\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[])$

Let $\mathcal{B} =$
$\{0;0;1;[],\ 0;1;[],\ 1;[],\ [],\ 0;1;0;1;[],\ 1;0;1;[]$

Constructing an algorithm

Rules:

$$\begin{array}{rcl rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Start term: $\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[])$

Let $\mathcal{B} =$

$\{0;0;1;[],\ 0;1;[],\ 1;[],\ [],\ 0;1;0;1;[],\ 1;0;1;[],\ \mathsf{True}$

Constructing an algorithm

Rules:

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s,\mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Start term: $\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[])$

Let $\mathcal{B} =$

$\{0;0;1;[],\ 0;1;[],\ 1;[],\ [],\ 0;1;0;1;[],\ 1;0;1;[],\ \mathsf{True},\ \mathsf{False}\}$

Constructing an algorithm

Rules:

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Start term: $\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[])$

Let $\mathcal{B} =$
$\{0;0;1;[],\ 0;1;[],\ 1;[],\ [],\ 0;1;0;1;[],\ 1;0;1;[],\ \mathsf{True},\ \mathsf{False}\}$

Make a list:

Constructing an algorithm

Rules:

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Start term: $\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[])$

Let $\mathcal{B} =$
$\{0;0;1;[],\ 0;1;[],\ 1;[],\ [],\ 0;1;0;1;[],\ 1;0;1;[],\ \mathsf{True},\ \mathsf{False}\}$

Make a list:

$$\mathsf{Subseq}(0;0;1;[],\ 0;0;1;[]) \quad \to^* \quad \{\}$$

Constructing an algorithm

Rules:

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Start term: $\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[])$

Let $\mathcal{B} =$

$\{0;0;1;[],\ 0;1;[],\ 1;[],\ [],\ 0;1;0;1;[],\ 1;0;1;[],\ \mathsf{True},\ \mathsf{False}\}$

Make a list:

$$\begin{array}{rcl} \mathsf{Subseq}(0;0;1;[],\ 0;0;1;[]) & \to^* & \{\} \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;[]) & \to^* & \{\} \end{array}$$

Constructing an algorithm

Rules:

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s,\mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Start term: $\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[])$

Let $\mathcal{B} =$
$\{0;0;1;[],\ 0;1;[],\ 1;[],\ [],\ 0;1;0;1;[],\ 1;0;1;[],\ \mathsf{True},\ \mathsf{False}\}$

Make a list:

$$\begin{array}{rcl} \mathsf{Subseq}(0;0;1;[],\ 0;0;1;[]) & \to^* & \{\} \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;[]) & \to^* & \{\} \\ & \dots & \end{array}$$

Constructing an algorithm

Rules:

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Start term: $\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[])$

Let $\mathcal{B} =$
$\{0;0;1;[],\ 0;1;[],\ 1;[],\ [],\ 0;1;0;1;[],\ 1;0;1;[],\ \mathsf{True},\ \mathsf{False}\}$

Make a list:

$$\begin{array}{rcl} \mathsf{Subseq}(0;0;1;[],\ 0;0;1;[]) & \to^* & \{\} \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;[]) & \to^* & \{\} \\ & \dots & \\ \mathsf{Subseq}(0;1;[],\ 0;0;1;[]) & \to^* & \{\} \end{array}$$

Constructing an algorithm

Rules:

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s,\mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Start term: $\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[])$

Let $\mathcal{B} =$

$\{0;0;1;[],\ 0;1;[],\ 1;[],\ [],\ 0;1;0;1;[],\ 1;0;1;[],\ \mathsf{True},\ \mathsf{False}\}$

Make a list:

$$\begin{array}{rcl} \mathsf{Subseq}(0;0;1;[],\ 0;0;1;[]) & \to^* & \{\} \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;[]) & \to^* & \{\} \\ & \dots & \\ \mathsf{Subseq}(0;1;[],\ 0;0;1;[]) & \to^* & \{\} \\ & \dots & \end{array}$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rl} \mathsf{tl}(0;1;0;1;[]) & \to^* \\ \mathsf{tl}(1;0;1;[]) & \to^* \\ \ldots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* \\ \mathsf{Subseq}([],\ []) & \to^* \\ \ldots & \end{array}$$

$$
\begin{array}{rcl@{\qquad}rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rl}
\mathsf{tl}(0;1;0;1;[]) & \to^* \\
\mathsf{tl}(1;0;1;[]) & \to^* \\
\dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* \\
\mathsf{Subseq}([],\ []) & \to^* \\
\dots &
\end{array}
$$

$$\begin{array}{rclrcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rl} \mathsf{tl}(0;1;0;1;[]) & \to^* \\ \mathsf{tl}(1;0;1;[]) & \to^* \\ \ldots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* \\ \mathsf{Subseq}([],\ []) & \to^* \\ \ldots & \end{array}$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\ \mathsf{Subseq}([],\ []) & \to^* & \\ & \dots & \end{array}$$

$$
\begin{array}{rclrcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & \\
 & \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\
\mathsf{Subseq}([],\ []) & \to^* & \\
 & \dots &
\end{array}
$$

$$
\begin{array}{rcl@{\qquad}rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & \\
& \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\
\mathsf{Subseq}([],\ []) & \to^* & \\
& \dots &
\end{array}
$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\ \mathsf{Subseq}([],\ []) & \to^* & \\ & \dots & \end{array}$$

$$
\begin{array}{rcl@{\qquad}rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[]
\end{array}
$$

...

$$
\begin{array}{rl}
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* \\
\mathsf{Subseq}([],\ []) & \to^*
\end{array}
$$

...

$$\begin{array}{rclrcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}$$

Statements:

$$\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
& \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\
\mathsf{Subseq}([],\ []) & \to^* & \\
& \dots &
\end{array}$$

$$
\begin{array}{rcl}
\color{red}{\mathsf{Subseq}([],\ t)} & \color{red}{\to} & \color{red}{\mathsf{True}} \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\
\mathsf{tl}(x;y) & \to & y
\end{array}
\qquad
\begin{array}{rcl}
\mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
& \dots & \\
\color{blue}{\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[])} & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\
\mathsf{Subseq}([],\ []) & \to^* & \\
& \dots &
\end{array}
$$

$$
\begin{array}{rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\
\mathsf{tl}(x;y) & \to & y
\end{array}
\qquad
\begin{array}{rcl}
\mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
 & \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\
\mathsf{Subseq}([],\ []) & \to^* & \\
 & \dots &
\end{array}
$$

$$
\begin{array}{rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\
\mathsf{tl}(x;y) & \to & y
\end{array}
\qquad
\begin{array}{rcl}
\mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
& \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\
\mathsf{Subseq}([],\ []) & \to^* & \\
& \dots &
\end{array}
$$

$$\begin{array}{rclrcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}$$

Statements:

$$\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
 & \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\
\mathsf{Subseq}([],\ []) & \to^* & \\
 & \dots &
\end{array}$$

$$
\begin{array}{rclrcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
& \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\
\mathsf{Subseq}([],\ []) & \to^* & \\
& \dots &
\end{array}
$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \ldots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\ \mathsf{Subseq}([],\ []) & \to^* & \\ & \ldots & \end{array}$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\ \mathsf{Subseq}([],\ []) & \to^* & \\ & \dots & \end{array}$$

$$\begin{array}{rclrcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}$$

Statements:

$$\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
& \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\
\mathsf{Subseq}([],\ []) & \to^* & \\
& \dots &
\end{array}$$

$$\begin{array}{rclrcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}$$

Statements:

$$\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
 & \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\
\mathsf{Subseq}([],\ []) & \to^* & \\
 & \dots &
\end{array}$$

$$
\begin{array}{rcl@{\qquad}rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
& \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\
\mathsf{Subseq}([],\ []) & \to^* & \\
& \dots &
\end{array}
$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s,\mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\ \mathsf{Subseq}([],\ []) & \to^* & \\ & \dots & \end{array}$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\ \mathsf{Subseq}([],\ []) & \to^* & \\ & \dots & \end{array}$$

$$
\begin{array}{rcl@{\qquad}rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
& \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\
\mathsf{Subseq}([],\ []) & \to^* & \\
& \dots &
\end{array}
$$

$$\begin{array}{rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\
\mathsf{tl}(x;y) & \to & y
\end{array}
\qquad
\begin{array}{rcl}
\mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}$$

Statements:

$$\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
 & \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\
\mathsf{Subseq}([],\ []) & \to^* & \\
 & \dots &
\end{array}$$

$$\begin{array}{rcl} \mathsf{Subseq}([], t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s, []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s, t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x, 0;t) & \to & \mathsf{Subseq}(s, t) \\ \mathsf{Subseq}(1;x, 1;t) & \to & \mathsf{Subseq}(s, t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \cdots & \\ \mathsf{Subseq}(0;0;1;[], 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[], 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[], []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[], 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[], 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[], 0;1;[]) & \to^* & \\ \mathsf{Subseq}(1;[], 1;[]) & \to^* & \\ \mathsf{Subseq}([], []) & \to^* & \\ & \cdots & \end{array}$$

$$
\begin{array}{rcl@{\quad}rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
& \ldots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\
\mathsf{Subseq}([],\ []) & \to^* & \\
& \ldots &
\end{array}
$$

$$
\begin{array}{rcl rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
 & \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\
\mathsf{Subseq}([],\ []) & \to^* & \\
 & \dots &
\end{array}
$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \quad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\ \mathsf{Subseq}([],\ []) & \to^* & \\ & \dots & \end{array}$$

$$
\begin{array}{rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\
\mathsf{tl}(x;y) & \to & y
\end{array}
\qquad
\begin{array}{rcl}
\mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
 & \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\
\mathsf{Subseq}([],\ []) & \to^* & \\
 & \dots &
\end{array}
$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\ \mathsf{Subseq}([],\ []) & \to^* & \\ & \dots & \end{array}$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \quad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s,\mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\ \mathsf{Subseq}([],\ []) & \to^* & \\ & \dots & \end{array}$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\ \mathsf{Subseq}([],\ []) & \to^* & \\ & \dots & \end{array}$$

$$
\begin{array}{rcl@{\qquad}rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s,\mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
& \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\
& \dots &
\end{array}
$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s,\mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True},\mathsf{False} \end{array}$$

...

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s,\mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \ldots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\ & \ldots & \end{array}$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s,\mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \color{blue}{\mathsf{tl}(1;0;1;[])} & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\ & \dots & \end{array}$$

$$
\begin{array}{rcl@{\qquad}rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
 & \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False}
\end{array}
$$

$\dots$

$$
\begin{array}{rcl@{\qquad}rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
 & \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\
 & \dots &
\end{array}
$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s,\mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\ & \dots & \end{array}$$

$$
\begin{array}{rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\
\mathsf{tl}(x;y) & \to & y
\end{array}
\qquad
\begin{array}{rcl}
\mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
& \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\color{blue}{\mathsf{Subseq}(0;0;1;[],\ [])} & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\
& \dots &
\end{array}
$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\ & \dots & \end{array}$$

$$
\begin{array}{rcl@{\qquad}rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
& \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\
& \dots &
\end{array}
$$

$$
\begin{array}{rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\
\mathsf{tl}(x;y) & \to & y
\end{array}
\qquad
\begin{array}{rcl}
\mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
 & \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\
 & \dots &
\end{array}
$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s,\mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True},\mathsf{False} \\ & \dots & \end{array}$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s,\mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \color{red}{\mathsf{Subseq}(1;x,\ 1;t)} & \color{red}{\to} & \color{red}{\mathsf{Subseq}(s,\ t)} \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\ \color{blue}{\mathsf{Subseq}(1;[],\ 1;[])} & \to^* & \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\ & \dots & \end{array}$$

$$
\begin{array}{rclrcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
& \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\
& \dots &
\end{array}
$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\ & \dots & \end{array}$$

$$
\begin{array}{rcl@{\qquad}rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
 & \ldots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\
 & \ldots &
\end{array}
$$

$$
\begin{array}{rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\
\mathsf{tl}(x;y) & \to & y
\end{array}
\qquad
\begin{array}{rcl}
\mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
 & \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\
 & \dots &
\end{array}
$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s,\mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True},\mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True},\mathsf{False} \\ & \dots & \end{array}$$

$$
\begin{array}{rcl@{\qquad}rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
 & \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False}
\end{array}
$$

...

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s,\mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True},\mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True},\mathsf{False} \\ & \dots & \end{array}$$

$$\begin{array}{rcl@{\qquad}rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}$$

Statements:

$$\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
& \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\
& \dots &
\end{array}$$

$$
\begin{array}{rcl@{\qquad}rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
 & \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False}
\end{array}
$$

...

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s,\mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\ & \dots & \end{array}$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s,\mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\ & \dots & \end{array}$$

$$
\begin{array}{rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\
\mathsf{tl}(x;y) & \to & y
\end{array}
\qquad
\begin{array}{rcl}
\mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
 & \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\color{blue}{\mathsf{Subseq}(0;0;1;[],\ [])} & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\
 & \dots &
\end{array}
$$

$$
\begin{array}{rcl@{\qquad}rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s,\mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
 & \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True},\mathsf{False} \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True},\mathsf{False} \\
 & \dots &
\end{array}
$$

$$
\begin{array}{rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\
\mathsf{tl}(x;y) & \to & y
\end{array}
\qquad
\begin{array}{rcl}
\mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
& \ldots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\
& \ldots &
\end{array}
$$

$$\begin{array}{rcl@{\qquad}rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}$$

Statements:

$$\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
& \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\
& \dots &
\end{array}$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\ & \dots & \end{array}$$

$$
\begin{array}{rcl@{\qquad}rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
& \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\color{blue}{\mathsf{Subseq}(1;[],\ 1;[])} & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\
& \dots &
\end{array}
$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s,\mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True},\mathsf{False} \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True},\mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True},\mathsf{False} \\ & \dots & \end{array}$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \end{array}$$

...

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\ & \dots & \end{array}$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \color{blue}{\mathsf{tl}(1;0;1;[])} & \to^* & 0;1;[] \\ & \ldots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\ & \ldots & \end{array}$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\ & \dots & \end{array}$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\ & \dots & \end{array}$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\ & \dots & \end{array}$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\ & \dots & \end{array}$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s,\mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\ & \dots & \end{array}$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s,\mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True},\mathsf{False} \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True},\mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True},\mathsf{False} \\ & \dots & \end{array}$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\ & \dots & \end{array}$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s,\mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\ & \dots & \end{array}$$

$$
\begin{array}{rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\
\mathsf{tl}(x;y) & \to & y
\end{array}
\qquad
\begin{array}{rcl}
\mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
 & \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\
 & \dots &
\end{array}
$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\ & \dots & \end{array}$$

$$
\begin{array}{rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\
\mathsf{tl}(x;y) & \to & y
\end{array}
\qquad
\begin{array}{rcl}
\mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
 & \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\
 & \dots &
\end{array}
$$

$$\begin{array}{rclrcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}$$

Statements:

$$\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
 & \ldots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False}
\end{array}$$

. . .

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\ & \dots & \end{array}$$

$$
\begin{array}{rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\
\mathsf{tl}(x;y) & \to & y
\end{array}
\qquad
\begin{array}{rcl}
\mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
 & \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\
 & \dots &
\end{array}
$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s,\mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True},\mathsf{False} \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True},\mathsf{False} \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True},\mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True},\mathsf{False} \end{array}$$

...

$$
\begin{array}{rcl@{\qquad}rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
 & \dots & \\
\color{blue}{\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[])} & \to^* & \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\
 & \dots &
\end{array}
$$

$$
\begin{array}{rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\
\mathsf{tl}(x;y) & \to & y
\end{array}
\qquad
\begin{array}{rcl}
\mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s,\mathsf{tl}(t)) \\
\mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
 & \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\
 & \dots &
\end{array}
$$

$$\begin{array}{rcl@{\qquad}rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}$$

Statements:

$$\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
 & \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\
 & \dots &
\end{array}$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \ldots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\ \color{blue}{\mathsf{Subseq}(0;0;1;[],\ [])} & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\ & \ldots & \end{array}$$

$$
\begin{array}{rcl@{\qquad}rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
& \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\
& \dots &
\end{array}
$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\ & \dots & \end{array}$$

$$
\begin{array}{rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\
\mathsf{tl}(x;y) & \to & y
\end{array}
\qquad
\begin{array}{rcl}
\mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
 & \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\
 & \dots &
\end{array}
$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \cdots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\ & \cdots & \end{array}$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\ & \dots & \end{array}$$

$$
\begin{array}{rcl@{\qquad}rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
 & \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\
 & \dots &
\end{array}
$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \end{array}$$

...

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s,\mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\ & \dots & \end{array}$$

$$
\begin{array}{rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\
\mathsf{tl}(x;y) & \to & y
\end{array}
\qquad
\begin{array}{rcl}
\mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s,\mathsf{tl}(t)) \\
\mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
 & \ldots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\
 & \ldots &
\end{array}
$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \end{array}$$

...

$$
\begin{array}{rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\
\mathsf{tl}(x;y) & \to & y
\end{array}
\qquad
\begin{array}{rcl}
\mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
& \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\
& \dots &
\end{array}
$$

$$
\begin{array}{rcl@{\qquad}rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
& \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\
& \dots &
\end{array}
$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \quad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\ \color{blue}{\mathsf{Subseq}(0;0;1;[],\ [])} & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\ & \dots & \end{array}$$

$$
\begin{array}{rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\
\mathsf{tl}(x;y) & \to & y
\end{array}
\qquad
\begin{array}{rcl}
\mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s,\mathsf{tl}(t)) \\
\mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
 & \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True},\mathsf{False} \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True},\mathsf{False} \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True},\mathsf{False} \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True},\mathsf{False} \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True},\mathsf{False} \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True},\mathsf{False} \\
 & \dots &
\end{array}
$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\ & \dots & \end{array}$$

$$
\begin{array}{rcl@{\qquad}rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
 & \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\
 & \dots &
\end{array}
$$

$$\begin{array}{rcl@{\qquad}rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}$$

Statements:

$$\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
 & \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\
 & \dots &
\end{array}$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s,\mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True},\mathsf{False} \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True},\mathsf{False} \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True},\mathsf{False} \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True},\mathsf{False} \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True},\mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True},\mathsf{False} \\ & \dots & \end{array}$$

$$\begin{array}{rclrcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}$$

Statements:

$$\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
& \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False}
\end{array}$$

. . .

$$
\begin{array}{rcl@{\qquad}rcl}
\mathsf{Subseq}([],\ t) & \to & \mathsf{True} & \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s, \mathsf{tl}(t)) \\
\mathsf{Subseq}(s,\ []) & \to & \mathsf{False} & \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\
\mathsf{tl}(x;y) & \to & y & \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t)
\end{array}
$$

Statements:

$$
\begin{array}{rcl}
\mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\
\mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\
 & \dots & \\
\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\
\mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\
\mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\
 & \dots &
\end{array}
$$

$$\begin{array}{rcl} \mathsf{Subseq}([],\ t) & \to & \mathsf{True} \\ \mathsf{Subseq}(s,\ []) & \to & \mathsf{False} \\ \mathsf{tl}(x;y) & \to & y \end{array} \qquad \begin{array}{rcl} \mathsf{Subseq}(s,\ t) & \to & \mathsf{Subseq}(s,\mathsf{tl}(t)) \\ \mathsf{Subseq}(0;x,\ 0;t) & \to & \mathsf{Subseq}(s,\ t) \\ \mathsf{Subseq}(1;x,\ 1;t) & \to & \mathsf{Subseq}(s,\ t) \end{array}$$

Statements:

$$\begin{array}{rcl} \mathsf{tl}(0;1;0;1;[]) & \to^* & 1;0;1;[] \\ \mathsf{tl}(1;0;1;[]) & \to^* & 0;1;[] \\ & \dots & \\ \color{red}{\mathsf{Subseq}(0;0;1;[],\ 0;1;0;1;[])} & \color{red}{\to^*} & \color{red}{\mathsf{True}}, \mathsf{False} \\ \mathsf{Subseq}(0;0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(0;0;1;[],\ []) & \to^* & \mathsf{False} \\ \mathsf{Subseq}(1;0;1;[],\ 0;1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(0;1;[],\ 1;0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(0;1;[],\ 0;1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}(1;[],\ 1;[]) & \to^* & \mathsf{True}, \mathsf{False} \\ \mathsf{Subseq}([],\ []) & \to^* & \mathsf{True}, \mathsf{False} \\ & \dots & \end{array}$$

We show:

- the decision problem accepted by a non-deterministic cons-free program with data order 0 (so a first-order cons-free TRS) is in P
- a non-deterministic cons-free program with data order 0 can decide any decision problem in P
- the decision problem accepted by a non-deterministic cons-free program is in ELEMENTARY regardless of data order
- a non-deterministic cons-free program with data order 1 can decide any decision problem in ELEMENTARY

We show:

- the decision problem accepted by a non-deterministic cons-free program with data order 0 (so a first-order cons-free TRS) is in P
- a non-deterministic cons-free program with data order 0 can decide any decision problem in P
- the decision problem accepted by a non-deterministic cons-free program is in ELEMENTARY regardless of data order
- a non-deterministic cons-free program with data order 1 can decide any decision problem in ELEMENTARY

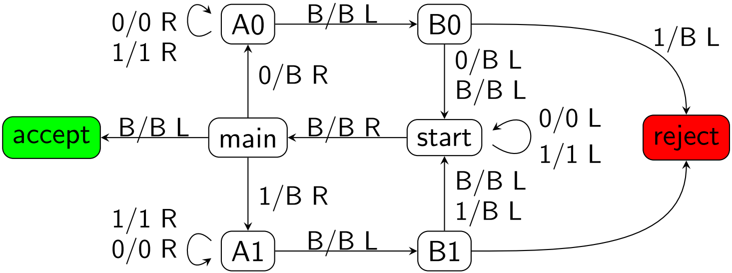
We show:

- the decision problem accepted by a non-deterministic cons-free program with data order 0 (so a first-order cons-free TRS) is in P
- a non-deterministic cons-free program with data order 0 can decide any decision problem in P
- the decision problem accepted by a non-deterministic cons-free program is in ELEMENTARY regardless of data order
- a non-deterministic cons-free program with data order 1 can decide any decision problem in ELEMENTARY

Claim: we can simulate any PTIME Turing Machine

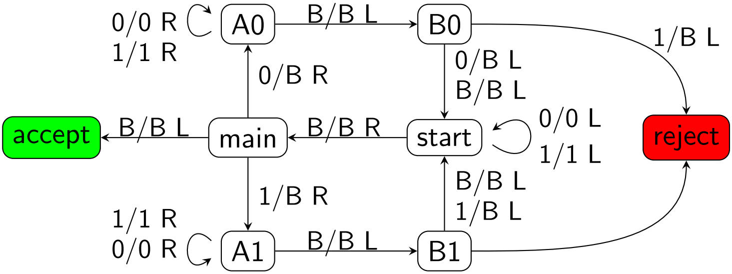
Claim: we can simulate any PTIME Turing Machine

Example:



Claim: we can simulate any PTIME Turing Machine

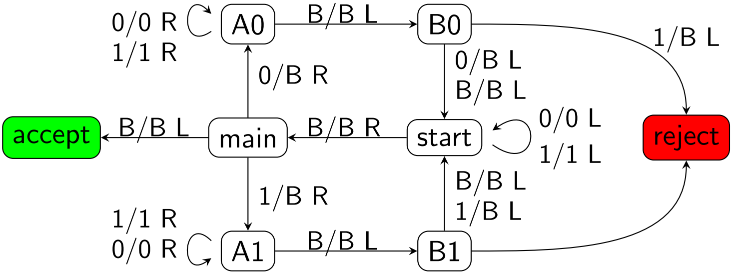
Example:



Runs in: $< 2 \cdot (n+1)^2$ steps

Claim: we can simulate any PTIME Turing Machine

Example:

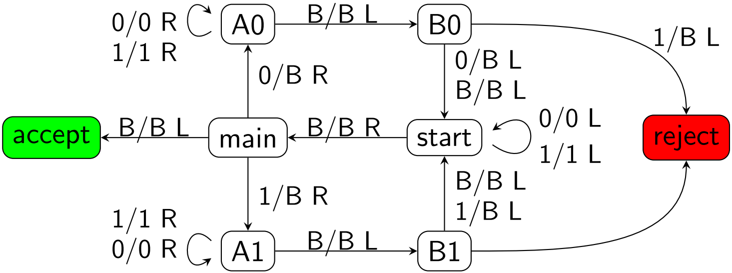


Runs in: $< 2 \cdot (n+1)^2$ steps

$$\mathsf{Transition}(\mathsf{Start}, 0) \quad \rightarrow \quad \mathsf{X}(\mathsf{Start}, 0, \mathsf{L})$$

Claim: we can simulate any PTIME Turing Machine

Example:

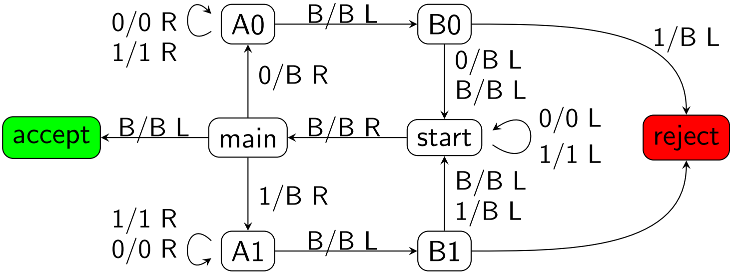


Runs in: $< 2 \cdot (n+1)^2$ steps

$$\begin{array}{lcl} \mathsf{Transition}(\mathsf{Start}, 0) & \to & \mathsf{X}(\mathsf{Start}, 0, \mathsf{L}) \\ \mathsf{Transition}(\mathsf{Start}, 1) & \to & \mathsf{X}(\mathsf{Start}, 1, \mathsf{L}) \\ \mathsf{Transition}(\mathsf{Start}, \mathsf{B}) & \to & \mathsf{X}(\mathsf{Main}, \mathsf{B}, \mathsf{R}) \end{array}$$

Claim: we can simulate any PTIME Turing Machine

Example:

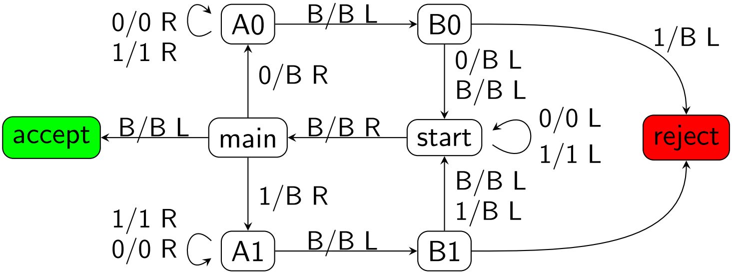


Runs in: $< 2 \cdot (n+1)^2$ steps

$$\begin{array}{lcl} \mathsf{Transition}(\mathsf{Start}, 0) & \rightarrow & \mathsf{X}(\mathsf{Start}, 0, \mathsf{L}) \\ \mathsf{Transition}(\mathsf{Start}, 1) & \rightarrow & \mathsf{X}(\mathsf{Start}, 1, \mathsf{L}) \\ \mathsf{Transition}(\mathsf{Start}, \mathsf{B}) & \rightarrow & \mathsf{X}(\mathsf{Main}, \mathsf{B}, \mathsf{R}) \\ \mathsf{Transition}(\mathsf{Main}, 0) & \rightarrow & \mathsf{X}(\mathsf{A0}, \mathsf{B}, \mathsf{R}) \end{array}$$

Claim: we can simulate any PTIME Turing Machine

Example:

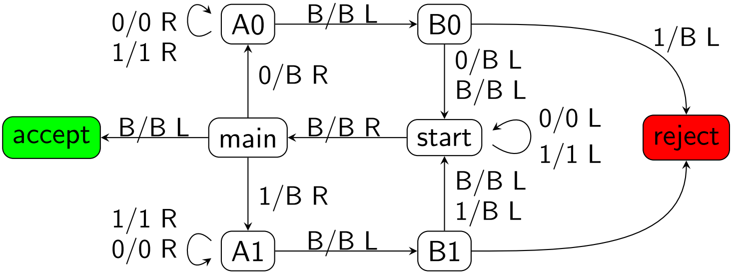


Runs in: $< 2 \cdot (n+1)^2$ steps

$$
\begin{array}{rcl}
\mathsf{Transition}(\mathsf{Start}, 0) & \to & \mathsf{X}(\mathsf{Start}, 0, \mathsf{L}) \\
\mathsf{Transition}(\mathsf{Start}, 1) & \to & \mathsf{X}(\mathsf{Start}, 1, \mathsf{L}) \\
\mathsf{Transition}(\mathsf{Start}, \mathsf{B}) & \to & \mathsf{X}(\mathsf{Main}, \mathsf{B}, \mathsf{R}) \\
\mathsf{Transition}(\mathsf{Main}, 0) & \to & \mathsf{X}(\mathsf{A0}, \mathsf{B}, \mathsf{R}) \\
\mathsf{Transition}(\mathsf{Accept}, x) & \to & \mathsf{X}(\mathsf{Accept}, x, \mathsf{N})
\end{array}
$$

Claim: we can simulate any PTIME Turing Machine

Example:



Runs in: $< 2 \cdot (n+1)^2$ steps

$$
\begin{array}{rcl}
\mathsf{Transition}(\mathsf{Start}, 0) & \to & \mathsf{X}(\mathsf{Start}, 0, \mathsf{L}) \\
\mathsf{Transition}(\mathsf{Start}, 1) & \to & \mathsf{X}(\mathsf{Start}, 1, \mathsf{L}) \\
\mathsf{Transition}(\mathsf{Start}, \mathsf{B}) & \to & \mathsf{X}(\mathsf{Main}, \mathsf{B}, \mathsf{R}) \\
\mathsf{Transition}(\mathsf{Main}, 0) & \to & \mathsf{X}(\mathsf{A0}, \mathsf{B}, \mathsf{R}) \\
\mathsf{Transition}(\mathsf{Accept}, x) & \to & \mathsf{X}(\mathsf{Accept}, x, \mathsf{N}) \\
& \ldots &
\end{array}
$$

Simulation: Turing Machine running in $< 2 \cdot (n+1)^2$ steps.

$$\begin{array}{rcl} \mathsf{Transition}(\mathsf{Start}, 0) & \to & \mathsf{X}(\mathsf{Start}, 0, \mathsf{L}) \\ \mathsf{Transition}(\mathsf{Start}, 1) & \to & \mathsf{X}(\mathsf{Start}, 1, \mathsf{L}) \\ \mathsf{Transition}(\mathsf{Start}, \mathsf{B}) & \to & \mathsf{X}(\mathsf{Main}, \mathsf{B}, \mathsf{R}) \\ \mathsf{Transition}(\mathsf{Main}, 0) & \to & \mathsf{X}(\mathsf{A0}, \mathsf{B}, \mathsf{R}) \\ \mathsf{Transition}(\mathsf{Accept}, x) & \to & \mathsf{X}(\mathsf{Accept}, x, \mathsf{S}) \\ & \dots & \end{array}$$

Simulation: Turing Machine running in $< 2 \cdot (n+1)^2$ steps.

$$\begin{array}{rcl} \mathsf{Transition}(\mathsf{Start}, 0) & \to & \mathsf{X}(\mathsf{Start}, 0, \mathsf{L}) \\ \mathsf{Transition}(\mathsf{Start}, 1) & \to & \mathsf{X}(\mathsf{Start}, 1, \mathsf{L}) \\ \mathsf{Transition}(\mathsf{Start}, \mathsf{B}) & \to & \mathsf{X}(\mathsf{Main}, \mathsf{B}, \mathsf{R}) \\ \mathsf{Transition}(\mathsf{Main}, 0) & \to & \mathsf{X}(\mathsf{A0}, \mathsf{B}, \mathsf{R}) \\ \mathsf{Transition}(\mathsf{Accept}, x) & \to & \mathsf{X}(\mathsf{Accept}, x, \mathsf{S}) \\ & \dots & \end{array}$$

Representation: $(l_1, l_2, l_3) \Longrightarrow |l_1| \cdot (n+1)^2 + |l_2| \cdot (n+1) + |l_3|$

Simulation: Turing Machine running in $< 2 \cdot (n+1)^2$ steps.

$$
\begin{array}{rcl}
\mathsf{Transition}(\mathsf{Start}, 0) & \to & \mathsf{X}(\mathsf{Start}, 0, \mathsf{L}) \\
\mathsf{Transition}(\mathsf{Start}, 1) & \to & \mathsf{X}(\mathsf{Start}, 1, \mathsf{L}) \\
\mathsf{Transition}(\mathsf{Start}, \mathsf{B}) & \to & \mathsf{X}(\mathsf{Main}, \mathsf{B}, \mathsf{R}) \\
\mathsf{Transition}(\mathsf{Main}, 0) & \to & \mathsf{X}(\mathsf{A0}, \mathsf{B}, \mathsf{R}) \\
\mathsf{Transition}(\mathsf{Accept}, x) & \to & \mathsf{X}(\mathsf{Accept}, x, \mathsf{S}) \\
 & \dots &
\end{array}
$$

Representation: $(l_1, l_2, l_3) \Longrightarrow |l_1| \cdot (n+1)^2 + |l_2| \cdot (n+1) + |l_3|$

$$\mathsf{TransitionAt}(i, t_1, t_2, t_3) \quad \to$$

Simulation: Turing Machine running in $< 2 \cdot (n+1)^2$ steps.

$$
\begin{array}{rcl}
\mathsf{Transition}(\mathsf{Start},0) & \to & \mathsf{X}(\mathsf{Start},0,\mathsf{L}) \\
\mathsf{Transition}(\mathsf{Start},1) & \to & \mathsf{X}(\mathsf{Start},1,\mathsf{L}) \\
\mathsf{Transition}(\mathsf{Start},\mathsf{B}) & \to & \mathsf{X}(\mathsf{Main},\mathsf{B},\mathsf{R}) \\
\mathsf{Transition}(\mathsf{Main},0) & \to & \mathsf{X}(\mathsf{A0},\mathsf{B},\mathsf{R}) \\
\mathsf{Transition}(\mathsf{Accept},x) & \to & \mathsf{X}(\mathsf{Accept},x,\mathsf{S}) \\
 & \dots &
\end{array}
$$

Representation: $(l_1, l_2, l_3) \Longrightarrow |l_1| \cdot (n+1)^2 + |l_2| \cdot (n+1) + |l_3|$

$$
\begin{array}{rcl}
\mathsf{TransitionAt}(i,t_1,t_2,t_3) & \to & \mathsf{Transition}(\mathsf{State}(i,t_1,t_2,t_3), \\
 & & \qquad \mathsf{CurTape}(i,t_1,t_2,t_3))
\end{array}
$$

Simulation: Turing Machine running in $< 2 \cdot (n+1)^2$ steps.

$$
\begin{array}{rcl}
\mathsf{Transition}(\mathsf{Start}, 0) & \to & \mathsf{X}(\mathsf{Start}, 0, \mathsf{L}) \\
\mathsf{Transition}(\mathsf{Start}, 1) & \to & \mathsf{X}(\mathsf{Start}, 1, \mathsf{L}) \\
\mathsf{Transition}(\mathsf{Start}, \mathsf{B}) & \to & \mathsf{X}(\mathsf{Main}, \mathsf{B}, \mathsf{R}) \\
\mathsf{Transition}(\mathsf{Main}, 0) & \to & \mathsf{X}(\mathsf{A0}, \mathsf{B}, \mathsf{R}) \\
\mathsf{Transition}(\mathsf{Accept}, x) & \to & \mathsf{X}(\mathsf{Accept}, x, \mathsf{S}) \\
& \dots &
\end{array}
$$

Representation: $(l_1, l_2, l_3) \Longrightarrow |l_1| \cdot (n+1)^2 + |l_2| \cdot (n+1) + |l_3|$

$$
\begin{array}{rcl}
\mathsf{TransitionAt}(i, t_1, t_2, t_3) & \to & \mathsf{Transition}(\mathsf{State}(i, t_1, t_2, t_3), \\
& & \qquad \mathsf{CurTape}(i, t_1, t_2, t_3)) \\
\mathsf{State}(i, [], [], []) & \to &
\end{array}
$$

Simulation: Turing Machine running in $< 2 \cdot (n+1)^2$ steps.

$$
\begin{array}{rcl}
\mathsf{Transition}(\mathsf{Start}, 0) & \to & \mathsf{X}(\mathsf{Start}, 0, \mathsf{L}) \\
\mathsf{Transition}(\mathsf{Start}, 1) & \to & \mathsf{X}(\mathsf{Start}, 1, \mathsf{L}) \\
\mathsf{Transition}(\mathsf{Start}, \mathsf{B}) & \to & \mathsf{X}(\mathsf{Main}, \mathsf{B}, \mathsf{R}) \\
\mathsf{Transition}(\mathsf{Main}, 0) & \to & \mathsf{X}(\mathsf{A0}, \mathsf{B}, \mathsf{R}) \\
\mathsf{Transition}(\mathsf{Accept}, x) & \to & \mathsf{X}(\mathsf{Accept}, x, \mathsf{S}) \\
 & \dots &
\end{array}
$$

Representation: $(l_1, l_2, l_3) \Longrightarrow |l_1| \cdot (n+1)^2 + |l_2| \cdot (n+1) + |l_3|$

$$
\begin{array}{rcl}
\mathsf{TransitionAt}(i, t_1, t_2, t_3) & \to & \mathsf{Transition}(\mathsf{State}(i, t_1, t_2, t_3), \\
 & & \quad \mathsf{CurTape}(i, t_1, t_2, t_3)) \\
\mathsf{State}(i, [], [], []) & \to & \mathsf{Start}
\end{array}
$$

Simulation: Turing Machine running in $< 2 \cdot (n+1)^2$ steps.

$$
\begin{array}{rcl}
\mathsf{Transition}(\mathsf{Start}, 0) & \to & \mathsf{X}(\mathsf{Start}, 0, \mathsf{L}) \\
\mathsf{Transition}(\mathsf{Start}, 1) & \to & \mathsf{X}(\mathsf{Start}, 1, \mathsf{L}) \\
\mathsf{Transition}(\mathsf{Start}, \mathsf{B}) & \to & \mathsf{X}(\mathsf{Main}, \mathsf{B}, \mathsf{R}) \\
\mathsf{Transition}(\mathsf{Main}, 0) & \to & \mathsf{X}(\mathsf{A0}, \mathsf{B}, \mathsf{R}) \\
\mathsf{Transition}(\mathsf{Accept}, x) & \to & \mathsf{X}(\mathsf{Accept}, x, \mathsf{S}) \\
 & \dots &
\end{array}
$$

Representation: $(l_1, l_2, l_3) \Longrightarrow |l_1| \cdot (n+1)^2 + |l_2| \cdot (n+1) + |l_3|$

$$
\begin{array}{rcl}
\mathsf{TransitionAt}(i, t_1, t_2, t_3) & \to & \mathsf{Transition}(\mathsf{State}(i, t_1, t_2, t_3), \\
 & & \qquad \mathsf{CurTape}(i, t_1, t_2, t_3)) \\
\mathsf{State}(i, [], [], []) & \to & \mathsf{Start} \\
\mathsf{State}(i, t_1, t_2, x; y) & \to & \\
\mathsf{State}(i, t_1, x; y, []) & \to & \\
\mathsf{State}(i, x; y, [], []) & \to &
\end{array}
$$

Simulation: Turing Machine running in $< 2 \cdot (n+1)^2$ steps.

$$
\begin{array}{rcl}
\mathsf{Transition}(\mathsf{Start}, 0) & \to & \mathsf{X}(\mathsf{Start}, 0, \mathsf{L}) \\
\mathsf{Transition}(\mathsf{Start}, 1) & \to & \mathsf{X}(\mathsf{Start}, 1, \mathsf{L}) \\
\mathsf{Transition}(\mathsf{Start}, \mathsf{B}) & \to & \mathsf{X}(\mathsf{Main}, \mathsf{B}, \mathsf{R}) \\
\mathsf{Transition}(\mathsf{Main}, 0) & \to & \mathsf{X}(\mathsf{A0}, \mathsf{B}, \mathsf{R}) \\
\mathsf{Transition}(\mathsf{Accept}, x) & \to & \mathsf{X}(\mathsf{Accept}, x, \mathsf{S}) \\
 & \dots &
\end{array}
$$

Representation: $(l_1, l_2, l_3) \Longrightarrow |l_1| \cdot (n+1)^2 + |l_2| \cdot (n+1) + |l_3|$

$$
\begin{array}{rcl}
\mathsf{TransitionAt}(i, t_1, t_2, t_3) & \to & \mathsf{Transition}(\mathsf{State}(i, t_1, t_2, t_3), \\
 & & \qquad \mathsf{CurTape}(i, t_1, t_2, t_3)) \\
\mathsf{State}(i, [], [], []) & \to & \mathsf{Start} \\
\mathsf{State}(i, t_1, t_2, x;y) & \to & \\
\mathsf{State}(i, t_1, x;y, []) & \to & \\
\mathsf{State}(i, x;y, [], []) & \to &
\end{array}
$$

Simulation: Turing Machine running in $< 2 \cdot (n+1)^2$ steps.

$$
\begin{array}{rcl}
\mathsf{Transition}(\mathsf{Start}, 0) & \to & \mathsf{X}(\mathsf{Start}, 0, \mathsf{L}) \\
\mathsf{Transition}(\mathsf{Start}, 1) & \to & \mathsf{X}(\mathsf{Start}, 1, \mathsf{L}) \\
\mathsf{Transition}(\mathsf{Start}, \mathsf{B}) & \to & \mathsf{X}(\mathsf{Main}, \mathsf{B}, \mathsf{R}) \\
\mathsf{Transition}(\mathsf{Main}, 0) & \to & \mathsf{X}(\mathsf{A0}, \mathsf{B}, \mathsf{R}) \\
\mathsf{Transition}(\mathsf{Accept}, x) & \to & \mathsf{X}(\mathsf{Accept}, x, \mathsf{S}) \\
 & \dots &
\end{array}
$$

Representation: $(l_1, l_2, l_3) \Longrightarrow |l_1| \cdot (n+1)^2 + |l_2| \cdot (n+1) + |l_3|$

$$
\begin{array}{rcl}
\mathsf{TransitionAt}(i, t_1, t_2, t_3) & \to & \mathsf{Transition}(\mathsf{State}(i, t_1, t_2, t_3), \\
 & & \qquad \mathsf{CurTape}(i, t_1, t_2, t_3)) \\
\mathsf{State}(i, [], [], []) & \to & \mathsf{Start} \\
\mathsf{State}(i, t_1, t_2, x; y) & \to & \mathsf{Fst}(\mathsf{TransitionAt}(i, t_1, t_2, y)) \\
\mathsf{State}(i, t_1, x; y, []) & \to & \mathsf{Fst}(\mathsf{TransitionAt}(i, t_1, y, i)) \\
\mathsf{State}(i, x; y, [], []) & \to & \mathsf{Fst}(\mathsf{TransitionAt}(i, y, i, i))
\end{array}
$$

Observation:

Observation:

to simulate a machine running in $< f(n)$ steps,
we must only be able to represent numbers $0, \dots, f(n) - 1$

Observation:

to simulate a machine running in $< f(n)$ steps,
we must only be able to represent numbers $0, \dots, f(n) - 1$

Basic Idea:

- simulate numbers $< A \cdot (n+1)^B$ by tuples $(l_1, \dots, l_B, l_{B+1})$

We show:

- the decision problem accepted by a non-deterministic cons-free program with data order 0 (so a first-order cons-free TRS) is in P
- a non-deterministic cons-free program with data order 0 can decide any decision problem in P
- the decision problem accepted by a non-deterministic cons-free program is in ELEMENTARY regardless of data order
- a non-deterministic cons-free program with data order 1 can decide any decision problem in ELEMENTARY

We show:

- the decision problem accepted by a non-deterministic cons-free program with data order 0 (so a first-order cons-free TRS) is in P
- a non-deterministic cons-free program with data order 0 can decide any decision problem in P
- the decision problem accepted by a non-deterministic cons-free program is in ELEMENTARY regardless of data order
- a non-deterministic cons-free program with data order 1 can decide any decision problem in ELEMENTARY

Observation:

to simulate a machine running in $< f(n)$ steps,
we must only be able to represent numbers $0, \ldots, f(n) - 1$

Basic Idea:

- simulate numbers $< A \cdot (n+1)^B$ by tuples $(l_1, \ldots, l_B, l_{B+1})$

Observation:

to simulate a machine running in $< f(n)$ steps,
we must only be able to represent numbers $0, \dots, f(n) - 1$

Basic Idea:

- simulate numbers $< A \cdot (n+1)^B$ by tuples $(l_1, \dots, l_B, l_{B+1})$

Logical Conclusion:

if we can count up to $\exp_2^k(n)$ for any k

Observation:

to simulate a machine running in $< f(n)$ steps,
we must only be able to represent numbers $0, \ldots, f(n) - 1$

Basic Idea:

- simulate numbers $< A \cdot (n+1)^B$ by tuples $(l_1, \ldots, l_B, l_{B+1})$

Logical Conclusion:

if we can count up to $\exp_2^k(n)$ for any k
then we can simulate any TM running in bounded time

Observation:

to simulate a machine running in $< f(n)$ steps,
we must only be able to represent numbers $0, \dots, f(n) - 1$

Basic Idea:

- simulate numbers $< A \cdot (n+1)^B$ by tuples $(l_1, \dots, l_B, l_{B+1})$

Logical Conclusion:

if we can count up to $\exp_2^k(n)$ for any k
then we can simulate any TM running in bounded time
so we can handle all problems in ELEMENTARY

Non-deterministic Counting:

Idea:

Non-deterministic Counting:

Idea:

- input list of length n, sublists represent numbers $0, \dots, n$

Non-deterministic Counting:

Idea:

- input list of length n, sublists represent numbers $0, \dots, n$
- term s : bool ⇒ list, represents bit vector $x_1 \dots x_n$ if:

Non-deterministic Counting:

Idea:

- input list of length n, sublists represent numbers $0, \dots, n$
- term s : bool ⇒ list, represents bit vector $x_1 \dots x_n$ if:
  - s True $\to^*$ "i" iff $x_i = 1$

Non-deterministic Counting:

Idea:

- input list of length n, sublists represent numbers $0, \dots, n$
- term s : bool ⇒ list, represents bit vector $x_1 \dots x_n$ if:
  - s True $\to^*$ "i" iff $x_i = 1$
  - s False $\to^*$ "i" iff $x_i = 0$

Non-deterministic Counting:

Idea:

- input list of length n, sublists represent numbers $0, \dots, n$
- term s : bool ⇒ list, represents bit vector $x_1 \dots x_n$ if:
  - s True $\to^*$ "i" iff $x_i = 1$
  - s False $\to^*$ "i" iff $x_i = 0$

Non-deterministic Counting:

Idea:

- input list of length n, sublists represent numbers $0, \dots, n$
- term s : bool ⇒ list, represents bit vector $x_1 \dots x_n$ if:
  - s True $\to^*$ "i" iff $x_i = 1$
  - s False $\to^*$ "i" iff $x_i = 0$

  use non-determinism!

Non-deterministic Counting:

Idea:

- input list of length n, sublists represent numbers $0, \dots, n$
- term s : bool ⇒ list, represents bit vector $x_1 \dots x_n$ if:
  - s True $\to^*$ "i" iff $x_i = 1$
  - s False $\to^*$ "i" iff $x_i = 0$

  use non-determinism!

set1 n F True	$\to$	n	set0 n F True	$\to$	F True
set1 n F True	$\to$	F True	set0 n F False	$\to$	n
set1 n F False	$\to$	F False	set0 n F False	$\to$	F False

Non-deterministic Counting:

Idea:

- input list of length n, sublists represent numbers $0, \ldots, n$
- term s : bool ⇒ list, represents bit vector $x_1 \ldots x_n$ if:
  - s True $\to^*$ "i" iff $x_i = 1$
  - s False $\to^*$ "i" iff $x_i = 0$

  use non-determinism!

set1 n F True	$\to$	n	set0 n F True	$\to$	F True
set1 n F True	$\to$	F True	set0 n F False	$\to$	n
set1 n F False	$\to$	F False	set0 n F False	$\to$	F False

bit vector 10110:

set1 "1" (set0 "2" (set1 "3" (set1 "4" (set0 "5" (const "0")))))

Non-deterministic Counting:

Idea:

- input list of length n, sublists represent numbers $0, \dots, n$
- term s : bool $\Rightarrow$ list, represents bit vector $x_1 \dots x_n$ if:
  - s True $\to^*$ "i" iff $x_i = 1$
  - s False $\to^*$ "i" iff $x_i = 0$

use non-determinism!

$$\begin{array}{lcl} \text{bitset } F\ n & \to & \text{if (equal } (F \text{ True})\ n) \text{ then True} \\ & & \text{else if (equal } (F \text{ False})\ n) \text{ then False} \\ & & \text{else bitset } F\ n \end{array}$$

Non-deterministic Counting:

Idea:

- input list of length n, sublists represent numbers $0, \dots, n$
- term s : bool $\Rightarrow$ list, represents bit vector $x_1 \dots x_n$ if:
  - s True $\rightarrow^*$ "i" iff $x_i = 1$
  - s False $\rightarrow^*$ "i" iff $x_i = 0$

Non-deterministic Counting:

Idea:

- input list of length n, sublists represent numbers $0, \dots, n$
- term s : bool ⇒ list, represents bit vector $x_1 \dots x_n$ if:
  - s True $\to^*$ "i" iff $x_i = 1$
  - s False $\to^*$ "i" iff $x_i = 0$
- term s : bool ⇒ (bool ⇒ list), represents bit vector $x_1 \dots x_{2^n}$ if:
  - s True $\to^*$ $[i]$ iff $x_i = 1$
  - s False $\to^*$ $[i]$ iff $x_i = 0$

Non-deterministic Counting:

Idea:

- input list of length n, sublists represent numbers $0, \dots, n$
- term s : bool ⇒ list, represents bit vector $x_1 \dots x_n$ if:
  - s True $\rightarrow^*$ "i" iff $x_i = 1$
  - s False $\rightarrow^*$ "i" iff $x_i = 0$
- term s : bool ⇒ (bool ⇒ list), represents bit vector $x_1 \dots x_{2^n}$ if:
  - s True $\rightarrow^*$ $[i]$ iff $x_i = 1$
  - s False $\rightarrow^*$ $[i]$ iff $x_i = 0$
- term s : bool ⇒ (bool ⇒ (bool ⇒ list)), represents bit vector $x_1 \dots x_{2^{2^n}}$ if:
  - s True $\rightarrow^*$ $[\![i]\!]$ iff $x_i = 1$
  - s False $\rightarrow^*$ $[\![i]\!]$ iff $x_i = 0$

Non-deterministic Counting:

Idea:

- input list of length n, sublists represent numbers $0, \dots, n$
- term s : bool $\Rightarrow$ list, represents bit vector $x_1 \dots x_n$ if:
  - s True $\to^*$ "i" iff $x_i = 1$
  - s False $\to^*$ "i" iff $x_i = 0$
- term s : bool $\Rightarrow$ (bool $\Rightarrow$ list), represents bit vector $x_1 \dots x_{2^n}$ if:
  - s True $\to^*$ $[i]$ iff $x_i = 1$
  - s False $\to^*$ $[i]$ iff $x_i = 0$
- term s : bool $\Rightarrow$ (bool $\Rightarrow$ (bool $\Rightarrow$ list)), represents bit vector $x_1 \dots x_{2^{2^n}}$ if:
  - s True $\to^*$ $[\![i]\!]$ iff $x_i = 1$
  - s False $\to^*$ $[\![i]\!]$ iff $x_i = 0$
- ...

We show:

- the decision problem accepted by a non-deterministic cons-free program with data order 0 (so a first-order cons-free TRS) is in P
- a non-deterministic cons-free program with data order 0 can decide any decision problem in P
- the decision problem accepted by a non-deterministic cons-free program is in ELEMENTARY regardless of data order
- a non-deterministic cons-free program with data order 1 can decide any decision problem in ELEMENTARY

Overview

Originally known:

Originally known:

- deterministic cons-free programs with data order k characterise $\mathsf{EXPTIME}^k$

Originally known:

- deterministic cons-free programs with data order k characterise $\mathsf{EXPTIME}^k$
- non-deterministic cons-free programs with data order 0 characterise $\mathsf{EXPTIME}^0$

Originally known:

- deterministic cons-free programs with data order k characterise $\mathrm{EXPTIME}^k$
- non-deterministic cons-free programs with data order 0 characterise $\mathrm{EXPTIME}^0$

What have we learned?

Originally known:

- deterministic cons-free programs with data order k characterise $\mathsf{EXPTIME}^k$
- non-deterministic cons-free programs with data order 0 characterise $\mathsf{EXPTIME}^0$

What have we learned?

- non-deterministic cons-free programs with data order k do not in general characterise $\mathsf{EXPTIME}^k$

Originally known:

- deterministic cons-free programs with data order k characterise $\mathsf{EXPTIME}^k$
- non-deterministic cons-free programs with data order 0 characterise $\mathsf{EXPTIME}^0$

What have we learned?

- non-deterministic cons-free programs with data order k do not in general characterise $\mathsf{EXPTIME}^k$
- non-deterministic cons-free programs with data order $k > 0$ characterise ELEMENTARY

Originally known:

- deterministic cons-free programs with data order k characterise EXPTIME^k
- non-deterministic cons-free programs with data order 0 characterise EXPTIME^0

What have we learned?

- non-deterministic cons-free programs with data order k do not in general characterise EXPTIME^k
- non-deterministic cons-free programs with data order $k > 0$ characterise ELEMENTARY

Why is this important?

Originally known:

- deterministic cons-free programs with data order k characterise EXPTIME^k
- non-deterministic cons-free programs with data order 0 characterise EXPTIME^0

What have we learned?

- non-deterministic cons-free programs with data order k do not in general characterise EXPTIME^k
- non-deterministic cons-free programs with data order $k > 0$ characterise ELEMENTARY

Why is this important?

- surprising result on the power of non-determinism

Originally known:

- deterministic cons-free programs with data order k characterise EXPTIME^k
- non-deterministic cons-free programs with data order 0 characterise EXPTIME^0

What have we learned?

- non-deterministic cons-free programs with data order k do not in general characterise EXPTIME^k
- non-deterministic cons-free programs with data order $k > 0$ characterise ELEMENTARY

Why is this important?

- surprising result on the power of non-determinism
- idea may be reusable towards more fine-grained characterisations

Originally known:

- deterministic cons-free programs with data order k characterise EXPTIME^k
- non-deterministic cons-free programs with data order 0 characterise EXPTIME^0

What have we learned?

- non-deterministic cons-free programs with data order k do not in general characterise EXPTIME^k
- non-deterministic cons-free programs with data order $k > 0$ characterise ELEMENTARY

Why is this important?

- surprising result on the power of non-determinism
- idea may be reusable towards more fine-grained characterisations

Questions?