

On First-order Cons-free Term Rewriting and PTIME

Cynthia Kop

2 Apr, 2016

Overview

- ① Motivation
- ② First-order Term Rewriting
- ③ Characterising PTIME
- ④ Call-by-value Rewriting
- ⑤ Other Restrictions
- ⑥ Conclusions

Overview

- 1 Motivation
- 2 First-order Term Rewriting
- 3 Characterising PTIME
- 4 Call-by-value Rewriting
- 5 Other Restrictions
- 6 Conclusions

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 $\mathcal{O}(\exp_2^k(n))$
- showed that arbitrary programs of data order k can be simulated by 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 $\mathcal{O}(\exp_2^k(n))$
- showed that arbitrary programs of data order k can be simulated by deterministic Turing Machines in $\mathcal{O}(\exp_2^k(n))$
- \implies 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 $\mathcal{O}(\exp_2^k(n))$
- showed that arbitrary programs of data order k can be simulated by deterministic Turing Machines in $\mathcal{O}(\exp_2^k(n))$
- \implies proved that programs of data order k *characterise* EXPTIME^k
- (proved several other characterisations as well)

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 $\mathcal{O}(\exp_2^k(n))$
- showed that arbitrary programs of data order k can be simulated by deterministic Turing Machines in $\mathcal{O}(\exp_2^k(n))$
- \implies proved that programs of data order k characterise EXPTIME^k
- (proved several other characterisations as well)

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 $\mathcal{O}(\exp_2^k(n))$
- showed that arbitrary programs of data order k can be simulated by deterministic Turing Machines in $\mathcal{O}(\exp_2^k(n))$
- \implies proved that programs of data order k characterise EXPTIME^k
- (proved several other characterisations as well)

Why not term rewriting?

Term rewriting:

Term rewriting:

- is the core calculus underlying functional programming

Term rewriting:

- is the core calculus underlying functional programming
- has a small definition (including higher-order versions)

Term rewriting:

- is the core calculus underlying functional programming
- has a small definition (including higher-order versions)
- already avoids complications present in real-world languages

Term rewriting:

- is the core calculus underlying functional programming
- has a small definition (including higher-order versions)
- already avoids complications present in real-world languages
- natively includes non-determinism

Term rewriting:

- is the core calculus underlying functional programming
- has a small definition (including higher-order versions)
- already avoids complications present in real-world languages
- natively includes non-determinism
- is supported by a large community with decades of theory

Term rewriting:

- is the core calculus underlying functional programming
- has a small definition (including higher-order versions)
- already avoids complications present in real-world languages
- natively includes non-determinism
- is supported by a large community with decades of theory
- allows for fine-tuning in many ways

Term rewriting:

- is the core calculus underlying functional programming
- has a small definition (including higher-order versions)
- already avoids complications present in real-world languages
- natively includes non-determinism
- is supported by a large community with decades of theory
- allows for fine-tuning in many ways

Let's see what comes out!

Overview

- 1 Motivation
- 2 First-order Term Rewriting**
- 3 Characterising PTIME
- 4 Call-by-value Rewriting
- 5 Other Restrictions
- 6 Conclusions

Terms: trees built from variables and function symbols.

Rules: generate a relation between terms.

Terms: trees built from variables and function symbols.

Rules: generate a relation between terms.

Example:

Function symbols:

$$\begin{array}{ll} 0 & : \quad [] \Rightarrow \mathbb{N} \\ S & : \quad [N] \Rightarrow \mathbb{N} \\ \text{Add} & : \quad [N \times N] \Rightarrow \mathbb{N} \\ \text{Mul} & : \quad [N \times N] \Rightarrow \mathbb{N} \end{array}$$

Terms: trees built from variables and function symbols.

Rules: generate a relation between terms.

Example:

Function symbols:

$$\begin{array}{ll} 0 & : \quad [] \Rightarrow \mathbb{N} \\ S & : \quad [N] \Rightarrow \mathbb{N} \end{array} \quad \begin{array}{ll} \text{Add} & : \quad [N \times N] \Rightarrow \mathbb{N} \\ \text{Mul} & : \quad [N \times N] \Rightarrow \mathbb{N} \end{array}$$

Rules:

$$\begin{array}{ll} \text{Add}(x, 0) & \rightarrow x \\ \text{Add}(0, y) & \rightarrow y \\ \text{Add}(x, S(y)) & \rightarrow S(\text{Add}(x, y)) \\ \text{Mul}(x, 0) & \rightarrow 0 \\ \text{Mul}(x, S(y)) & \rightarrow \text{Add}(\text{Mul}(x, y), x) \end{array}$$

Terms: trees built from variables and function symbols.

Rules: generate a relation between terms.

Example:

Function symbols:

$$\begin{array}{ll} 0 & : \quad [] \Rightarrow \mathbb{N} \\ S & : \quad [\mathbb{N}] \Rightarrow \mathbb{N} \end{array} \quad \begin{array}{ll} \text{Add} & : \quad [\mathbb{N} \times \mathbb{N}] \Rightarrow \mathbb{N} \\ \text{Mul} & : \quad [\mathbb{N} \times \mathbb{N}] \Rightarrow \mathbb{N} \end{array}$$

Rules:

$$\begin{array}{ll} \text{Add}(x, 0) & \rightarrow x \\ \text{Add}(0, y) & \rightarrow y \\ \text{Add}(x, S(y)) & \rightarrow S(\text{Add}(x, y)) \\ \text{Mul}(x, 0) & \rightarrow 0 \\ \text{Mul}(x, S(y)) & \rightarrow \text{Add}(\text{Mul}(x, y), x) \end{array}$$

Example reduction:

$\text{Add}(S(0), S(0))$

Terms: trees built from variables and function symbols.

Rules: generate a relation between terms.

Example:

Function symbols:

$$\begin{array}{ll} 0 & : \quad [] \Rightarrow \mathbb{N} \\ S & : \quad [N] \Rightarrow \mathbb{N} \end{array} \quad \begin{array}{ll} \text{Add} & : \quad [N \times N] \Rightarrow \mathbb{N} \\ \text{Mul} & : \quad [N \times N] \Rightarrow \mathbb{N} \end{array}$$

Rules:

$$\begin{array}{ll} \text{Add}(x, 0) & \rightarrow x \\ \text{Add}(0, y) & \rightarrow y \\ \text{Add}(x, S(y)) & \rightarrow S(\text{Add}(x, y)) \\ \text{Mul}(x, 0) & \rightarrow 0 \\ \text{Mul}(x, S(y)) & \rightarrow \text{Add}(\text{Mul}(x, y), x) \end{array}$$

Example reduction:

$\text{Add}(S(0), S(0))$

Terms: trees built from variables and function symbols.

Rules: generate a relation between terms.

Example:

Function symbols:

$$\begin{array}{ll} 0 & : \quad [] \Rightarrow \mathbb{N} \\ S & : \quad [N] \Rightarrow \mathbb{N} \end{array} \quad \begin{array}{ll} \text{Add} & : \quad [N \times N] \Rightarrow \mathbb{N} \\ \text{Mul} & : \quad [N \times N] \Rightarrow \mathbb{N} \end{array}$$

Rules:

$$\begin{array}{ll} \text{Add}(x, 0) & \rightarrow x \\ \text{Add}(0, y) & \rightarrow y \\ \text{Add}(x, S(y)) & \rightarrow S(\text{Add}(x, y)) \\ \text{Mul}(x, 0) & \rightarrow 0 \\ \text{Mul}(x, S(y)) & \rightarrow \text{Add}(\text{Mul}(x, y), x) \end{array}$$

Example reduction:

$$\text{Add}(S(0), S(0)) \rightarrow S(\text{Add}(S(0), 0))$$

Terms: trees built from variables and function symbols.

Rules: generate a relation between terms.

Example:

Function symbols:

$$\begin{array}{ll} 0 & : \quad [] \Rightarrow \mathbb{N} \\ S & : \quad [N] \Rightarrow \mathbb{N} \end{array} \quad \begin{array}{ll} \text{Add} & : \quad [N \times N] \Rightarrow \mathbb{N} \\ \text{Mul} & : \quad [N \times N] \Rightarrow \mathbb{N} \end{array}$$

Rules:

$$\begin{array}{ll} \text{Add}(x, 0) & \rightarrow x \\ \text{Add}(0, y) & \rightarrow y \\ \text{Add}(x, S(y)) & \rightarrow S(\text{Add}(x, y)) \\ \text{Mul}(x, 0) & \rightarrow 0 \\ \text{Mul}(x, S(y)) & \rightarrow \text{Add}(\text{Mul}(x, y), x) \end{array}$$

Example reduction:

$$\text{Add}(S(0), S(0)) \rightarrow S(\text{Add}(S(0), 0))$$

Terms: trees built from variables and function symbols.

Rules: generate a relation between terms.

Example:

Function symbols:

$$\begin{array}{ll} 0 & : \quad [] \Rightarrow \mathbb{N} \\ S & : \quad [N] \Rightarrow \mathbb{N} \end{array} \quad \begin{array}{ll} \text{Add} & : \quad [N \times N] \Rightarrow \mathbb{N} \\ \text{Mul} & : \quad [N \times N] \Rightarrow \mathbb{N} \end{array}$$

Rules:

$$\begin{array}{ll} \text{Add}(x, 0) & \rightarrow x \\ \text{Add}(0, y) & \rightarrow y \\ \text{Add}(x, S(y)) & \rightarrow S(\text{Add}(x, y)) \\ \text{Mul}(x, 0) & \rightarrow 0 \\ \text{Mul}(x, S(y)) & \rightarrow \text{Add}(\text{Mul}(x, y), x) \end{array}$$

Example reduction:

$$\text{Add}(S(0), S(0)) \rightarrow S(\text{Add}(S(0), 0))$$

Terms: trees built from variables and function symbols.

Rules: generate a relation between terms.

Example:

Function symbols:

$$\begin{array}{ll} 0 & : \quad [] \Rightarrow \mathbb{N} \\ S & : \quad [N] \Rightarrow \mathbb{N} \end{array} \quad \begin{array}{ll} \text{Add} & : \quad [N \times N] \Rightarrow \mathbb{N} \\ \text{Mul} & : \quad [N \times N] \Rightarrow \mathbb{N} \end{array}$$

Rules:

$$\begin{array}{ll} \text{Add}(x, 0) & \rightarrow x \\ \text{Add}(0, y) & \rightarrow y \\ \text{Add}(x, S(y)) & \rightarrow S(\text{Add}(x, y)) \\ \text{Mul}(x, 0) & \rightarrow 0 \\ \text{Mul}(x, S(y)) & \rightarrow \text{Add}(\text{Mul}(x, y), x) \end{array}$$

Example reduction:

$$\text{Add}(S(0), S(0)) \rightarrow S(\text{Add}(S(0), 0)) \rightarrow S(S(0))$$

Peculiarities

Note that:

Peculiarities

Note that:

- rule selection is non-deterministic (unless otherwise specified)

Peculiarities

Note that:

- rule selection is non-deterministic (unless otherwise specified)
- position selection is non-deterministic (unless otherwise specified)

Peculiarities

Note that:

- rule selection is non-deterministic (unless otherwise specified)
- position selection is non-deterministic (unless otherwise specified)

$$\text{Add}(x, 0) \rightarrow x$$

$$\text{Add}(0, y) \rightarrow y$$

$$\text{Add}(x, \text{S}(y)) \rightarrow \text{S}(\text{Add}(x, y))$$

$$\text{Mul}(x, 0) \rightarrow 0$$

$$\text{Mul}(x, \text{S}(y)) \rightarrow \text{Add}(\text{Mul}(x, y), x)$$

Example reduction:

$\text{Mul}(\text{S}(0), \text{S}(0))$

Peculiarities

Note that:

- rule selection is non-deterministic (unless otherwise specified)
- position selection is non-deterministic (unless otherwise specified)

$$\text{Add}(x, 0) \rightarrow x$$

$$\text{Add}(0, y) \rightarrow y$$

$$\text{Add}(x, \text{S}(y)) \rightarrow \text{S}(\text{Add}(x, y))$$

$$\text{Mul}(x, 0) \rightarrow 0$$

$$\text{Mul}(x, \text{S}(y)) \rightarrow \text{Add}(\text{Mul}(x, y), x)$$

Example reduction:

$$\text{Mul}(\text{S}(0), \text{S}(0))$$

Peculiarities

Note that:

- rule selection is non-deterministic (unless otherwise specified)
- position selection is non-deterministic (unless otherwise specified)

$$\text{Add}(x, 0) \rightarrow x$$

$$\text{Add}(0, y) \rightarrow y$$

$$\text{Add}(x, \text{S}(y)) \rightarrow \text{S}(\text{Add}(x, y))$$

$$\text{Mul}(x, 0) \rightarrow 0$$

$$\text{Mul}(x, \text{S}(y)) \rightarrow \text{Add}(\text{Mul}(x, y), x)$$

Example reduction:

$$\text{Mul}(\text{S}(0), \text{S}(0)) \rightarrow \text{Add}(\text{Mul}(\text{S}(0), 0), \text{S}(0))$$

Peculiarities

Note that:

- rule selection is non-deterministic (unless otherwise specified)
- position selection is non-deterministic (unless otherwise specified)

$$\text{Add}(x, 0) \rightarrow x$$

$$\text{Add}(0, y) \rightarrow y$$

$$\text{Add}(x, S(y)) \rightarrow S(\text{Add}(x, y))$$

$$\text{Mul}(x, 0) \rightarrow 0$$

$$\text{Mul}(x, S(y)) \rightarrow \text{Add}(\text{Mul}(x, y), x)$$

Example reduction:

$$\text{Mul}(S(0), S(0)) \rightarrow \text{Add}(\text{Mul}(S(0), 0), S(0))$$

Peculiarities

Note that:

- rule selection is non-deterministic (unless otherwise specified)
- position selection is non-deterministic (unless otherwise specified)

$$\begin{aligned}\text{Add}(x, 0) &\rightarrow x \\ \text{Add}(0, y) &\rightarrow y \\ \text{Add}(x, \text{S}(y)) &\rightarrow \text{S}(\text{Add}(x, y)) \\ \text{Mul}(x, 0) &\rightarrow 0 \\ \text{Mul}(x, \text{S}(y)) &\rightarrow \text{Add}(\text{Mul}(x, y), x)\end{aligned}$$

Example reduction:

$$\text{Mul}(\text{S}(0), \text{S}(0)) \rightarrow \text{Add}(\text{Mul}(\text{S}(0), 0), \text{S}(0))$$

Peculiarities

Note that:

- rule selection is non-deterministic (unless otherwise specified)
- position selection is non-deterministic (unless otherwise specified)

$$\text{Add}(x, 0) \rightarrow x$$

$$\text{Add}(0, y) \rightarrow y$$

$$\text{Add}(x, \text{S}(y)) \rightarrow \text{S}(\text{Add}(x, y))$$

$$\text{Mul}(x, 0) \rightarrow 0$$

$$\text{Mul}(x, \text{S}(y)) \rightarrow \text{Add}(\text{Mul}(x, y), x)$$

Example reduction:

$$\text{Mul}(\text{S}(0), \text{S}(0)) \rightarrow \text{Add}(\text{Mul}(\text{S}(0), 0), \text{S}(0))$$

Peculiarities

Note that:

- rule selection is non-deterministic (unless otherwise specified)
- position selection is non-deterministic (unless otherwise specified)

$$\text{Add}(x, 0) \rightarrow x$$

$$\text{Add}(0, y) \rightarrow y$$

$$\text{Add}(x, \text{S}(y)) \rightarrow \text{S}(\text{Add}(x, y))$$

$$\text{Mul}(x, 0) \rightarrow 0$$

$$\text{Mul}(x, \text{S}(y)) \rightarrow \text{Add}(\text{Mul}(x, y), x)$$

Example reduction:

$$\text{Mul}(\text{S}(0), \text{S}(0)) \rightarrow \text{Add}(\text{Mul}(\text{S}(0), 0), \text{S}(0)) \rightarrow \\ \text{S}(\text{Add}(\text{Mul}(\text{S}(0), 0), 0))$$

Peculiarities

Note that:

- rule selection is non-deterministic (unless otherwise specified)
- position selection is non-deterministic (unless otherwise specified)

$$\text{Add}(x, 0) \rightarrow x$$

$$\text{Add}(0, y) \rightarrow y$$

$$\text{Add}(x, \text{S}(y)) \rightarrow \text{S}(\text{Add}(x, y))$$

$$\text{Mul}(x, 0) \rightarrow 0$$

$$\text{Mul}(x, \text{S}(y)) \rightarrow \text{Add}(\text{Mul}(x, y), x)$$

Example reduction:

$$\begin{aligned} \text{Mul}(\text{S}(0), \text{S}(0)) &\rightarrow \text{Add}(\text{Mul}(\text{S}(0), 0), \text{S}(0)) \rightarrow \\ &\text{S}(\text{Add}(\text{Mul}(\text{S}(0), 0), 0)) \end{aligned}$$

Peculiarities

Note that:

- rule selection is non-deterministic (unless otherwise specified)
- position selection is non-deterministic (unless otherwise specified)

$$\text{Add}(x, 0) \rightarrow x$$

$$\text{Add}(0, y) \rightarrow y$$

$$\text{Add}(x, \text{S}(y)) \rightarrow \text{S}(\text{Add}(x, y))$$

$$\text{Mul}(x, 0) \rightarrow 0$$

$$\text{Mul}(x, \text{S}(y)) \rightarrow \text{Add}(\text{Mul}(x, y), x)$$

Example reduction:

$$\text{Mul}(\text{S}(0), \text{S}(0)) \rightarrow \text{Add}(\text{Mul}(\text{S}(0), 0), \text{S}(0)) \rightarrow \text{S}(\text{Add}(\text{Mul}(\text{S}(0), 0), 0))$$

Peculiarities

Note that:

- rule selection is non-deterministic (unless otherwise specified)
- position selection is non-deterministic (unless otherwise specified)

$$\begin{aligned}\text{Add}(x, 0) &\rightarrow x \\ \text{Add}(0, y) &\rightarrow y \\ \text{Add}(x, \text{S}(y)) &\rightarrow \text{S}(\text{Add}(x, y)) \\ \text{Mul}(x, 0) &\rightarrow \mathbf{0} \\ \text{Mul}(x, \text{S}(y)) &\rightarrow \text{Add}(\text{Mul}(x, y), x)\end{aligned}$$

Example reduction:

$$\begin{aligned}\text{Mul}(\text{S}(0), \text{S}(0)) &\rightarrow \text{Add}(\text{Mul}(\text{S}(0), 0), \text{S}(0)) \rightarrow \\ &\text{S}(\text{Add}(\text{Mul}(\text{S}(0), 0), 0)) \rightarrow \text{S}(\text{Add}(\mathbf{0}, 0))\end{aligned}$$

Peculiarities

Note that:

- rule selection is non-deterministic (unless otherwise specified)
- position selection is non-deterministic (unless otherwise specified)

$$\text{Add}(x, 0) \rightarrow x$$

$$\text{Add}(0, y) \rightarrow y$$

$$\text{Add}(x, S(y)) \rightarrow S(\text{Add}(x, y))$$

$$\text{Mul}(x, 0) \rightarrow 0$$

$$\text{Mul}(x, S(y)) \rightarrow \text{Add}(\text{Mul}(x, y), x)$$

Example reduction:

$$\begin{aligned} \text{Mul}(S(0), S(0)) &\rightarrow \text{Add}(\text{Mul}(S(0), 0), S(0)) \rightarrow \\ &S(\text{Add}(\text{Mul}(S(0), 0), 0)) \rightarrow S(\text{Add}(0, 0)) \end{aligned}$$

Overview

- 1 Motivation
- 2 First-order Term Rewriting
- 3 Characterising PTIME**
- 4 Call-by-value Rewriting
- 5 Other Restrictions
- 6 Conclusions

Extending cons-free

Extending cons-free

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

Extending cons-free

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

Extending cons-free

Function symbols:

$$\begin{array}{ll} 0 & : \quad [] \Rightarrow \mathbf{N} \quad \text{Add} & : \quad [\mathbf{N} \times \mathbf{N}] \Rightarrow \mathbf{N} \\ \mathbf{S} & : \quad [\mathbf{N}] \Rightarrow \mathbf{N} \quad \text{Mul} & : \quad [\mathbf{N} \times \mathbf{N}] \Rightarrow \mathbf{N} \end{array}$$

Rules:

$$\begin{array}{ll} \text{Add}(x, 0) & \rightarrow x \\ \text{Add}(0, y) & \rightarrow y \\ \text{Add}(x, \mathbf{S}(y)) & \rightarrow \mathbf{S}(\text{Add}(x, y)) \\ \text{Mul}(x, 0) & \rightarrow 0 \\ \text{Mul}(x, \mathbf{S}(y)) & \rightarrow \text{Add}(\text{Mul}(x, y), x) \end{array}$$

Extending cons-free

Function symbols:

$$\begin{array}{ll} 0 & : \quad [] \Rightarrow \mathbf{N} \quad \text{Add} & : \quad [\mathbf{N} \times \mathbf{N}] \Rightarrow \mathbf{N} \\ \mathbf{S} & : \quad [\mathbf{N}] \Rightarrow \mathbf{N} \quad \text{Mul} & : \quad [\mathbf{N} \times \mathbf{N}] \Rightarrow \mathbf{N} \end{array}$$

Rules:

$$\begin{array}{ll} \text{Add}(x, 0) & \rightarrow x \\ \text{Add}(0, y) & \rightarrow y \\ \text{Add}(x, \mathbf{S}(y)) & \rightarrow \mathbf{S}(\text{Add}(x, y)) \\ \text{Mul}(x, 0) & \rightarrow 0 \\ \text{Mul}(x, \mathbf{S}(y)) & \rightarrow \text{Add}(\text{Mul}(x, y), x) \end{array}$$

Extending cons-free

Function symbols:

$$\begin{array}{ll}
 0 & : \quad [] \Rightarrow \mathbf{N} & \text{Add} & : \quad [\mathbf{N} \times \mathbf{N}] \Rightarrow \mathbf{N} \\
 S & : \quad [\mathbf{N}] \Rightarrow \mathbf{N} & \text{Mul} & : \quad [\mathbf{N} \times \mathbf{N}] \Rightarrow \mathbf{N}
 \end{array}$$

Rules:

$$\begin{array}{ll}
 \text{Add}(x, 0) & \rightarrow x \\
 \text{Add}(0, y) & \rightarrow y \\
 \text{Add}(x, S(y)) & \rightarrow S(\text{Add}(x, y)) \\
 \text{Mul}(x, 0) & \rightarrow 0 \\
 \text{Mul}(x, S(y)) & \rightarrow \text{Add}(\text{Mul}(x, y), x)
 \end{array}$$

Extending cons-free

Function symbols:

$$\begin{array}{ll}
 0 & : \quad [] \Rightarrow \mathbf{N} & \text{Add} & : \quad [\mathbf{N} \times \mathbf{N}] \Rightarrow \mathbf{N} \\
 S & : \quad [\mathbf{N}] \Rightarrow \mathbf{N} & \text{Mul} & : \quad [\mathbf{N} \times \mathbf{N}] \Rightarrow \mathbf{N}
 \end{array}$$

Rules:

$$\begin{array}{ll}
 \text{Add}(x, 0) & \rightarrow x \\
 \text{Add}(0, y) & \rightarrow y \\
 \text{Add}(x, S(y)) & \rightarrow \underline{S}(\text{Add}(x, y)) \\
 \text{Mul}(x, 0) & \rightarrow 0 \\
 \text{Mul}(x, S(y)) & \rightarrow \text{Add}(\text{Mul}(x, y), x)
 \end{array}$$

Cons-free Term Rewriting:

- symbols are split into **data** and **function** symbols

Cons-free Term Rewriting:

- symbols are split into **data** and **function** symbols
- right-hand sides create no new **data**:

Cons-free Term Rewriting:

- symbols are split into **data** and **function** symbols
- right-hand sides create no new **data**:
 - right-hand sides *may* contain terms exclusively built of constructors (e.g. True, False, S(0))

Cons-free Term Rewriting:

- symbols are split into **data** and **function** symbols
- right-hand sides create no new **data**:
 - right-hand sides *may* contain terms exclusively built of constructors (e.g. True, False, S(0))
 - right-hand sides *may* contain subterms of their left-hand side (to allow for pattern matching)

Cons-free Term Rewriting:

- symbols are split into **data** and **function** symbols
- right-hand sides create no new **data**:
 - right-hand sides *may* contain terms exclusively built of constructors (e.g. True, False, S(0))
 - right-hand sides *may* contain subterms of their left-hand side (to allow for pattern matching)
 - right-hand sides *may not* contain data symbols otherwise

Cons-free Term Rewriting:

- symbols are split into **data** and **function** symbols
- right-hand sides create no new **data**:
 - right-hand sides *may* contain terms exclusively built of constructors (e.g. True, False, S(0))
 - right-hand sides *may* contain subterms of their left-hand side (to allow for pattern matching)
 - right-hand sides *may not* contain data symbols otherwise

$$\mathbf{Hd}(h;t) \rightarrow h$$

$$\mathbf{Tl}(h;t) \rightarrow t$$

Cons-free Term Rewriting:

- symbols are split into **data** and **function** symbols
- right-hand sides create no new **data**:
 - right-hand sides *may* contain terms exclusively built of constructors (e.g. True, False, S(0))
 - right-hand sides *may* contain subterms of their left-hand side (to allow for pattern matching)
 - right-hand sides *may not* contain data symbols otherwise

Evenlength(l) \rightarrow **Helper**(l , **True**)

Helper($[], b$) $\rightarrow b$

Helper($h;t$, **True**) \rightarrow **Helper**(t , **False**)

Helper($h;t$, **False**) \rightarrow **Helper**(t , **True**)

Cons-free Term Rewriting:

- symbols are split into **data** and **function** symbols
- right-hand sides create no new **data**:
 - right-hand sides *may* contain terms exclusively built of constructors (e.g. True, False, S(0))
 - right-hand sides *may* contain subterms of their left-hand side (to allow for pattern matching)
 - right-hand sides *may not* contain data symbols otherwise

$$\text{Half}(0) \rightarrow 0$$

$$\text{Half}(S(0)) \rightarrow 0$$

$$\text{Half}(S(S(x))) \rightarrow S(\text{Half}(x))$$

Cons-free Term Rewriting:

- symbols are split into **data** and **function** symbols
- right-hand sides create no new **data**:
 - right-hand sides *may* contain terms exclusively built of constructors (e.g. True, False, S(0))
 - right-hand sides *may* contain subterms of their left-hand side (to allow for pattern matching)
 - right-hand sides *may not* contain data symbols otherwise

$$\text{Half}(0) \rightarrow 0$$

$$\text{Half}(S(0)) \rightarrow 0$$

$$\text{Half}(S(S(x))) \rightarrow S(\text{Half}(x))$$

Cons-free Term Rewriting:

- symbols are split into **data** and **function** symbols
- right-hand sides create no new **data**:
 - right-hand sides *may* contain terms exclusively built of constructors (e.g. True, False, S(0))
 - right-hand sides *may* contain subterms of their left-hand side (to allow for pattern matching)
 - right-hand sides *may not* contain data symbols otherwise

$$\begin{aligned}
 \text{Half}(x) &\rightarrow \text{Helper}(x, x) \\
 \text{Helper}(0, y) &\rightarrow y \\
 \text{Helper}(S(0), S(y)) &\rightarrow y \\
 \text{Helper}(S(S(x)), S(y)) &\rightarrow \text{Helper}(x, y)
 \end{aligned}$$

Cons-free Term Rewriting:

- symbols are split into **data** and **function** symbols
- right-hand sides create no new **data**:
 - right-hand sides *may* contain terms exclusively built of constructors (e.g. True, False, S(0))
 - right-hand sides *may* contain subterms of their left-hand side (to allow for pattern matching)
 - right-hand sides *may not* contain data symbols otherwise

$$\text{Succ}(x) \rightarrow S(x)$$

Cons-free Term Rewriting:

- symbols are split into **data** and **function** symbols
- right-hand sides create no new **data**:
 - right-hand sides *may* contain terms exclusively built of constructors (e.g. True, False, S(0))
 - right-hand sides *may* contain subterms of their left-hand side (to allow for pattern matching)
 - right-hand sides *may not* contain data symbols otherwise

$$\text{Succ}(x) \rightarrow S(x)$$

Cons-free Term Rewriting:

- symbols are split into **data** and **function** symbols
- right-hand sides create no new **data**:
 - right-hand sides *may* contain terms exclusively built of constructors (e.g. True, False, S(0))
 - right-hand sides *may* contain subterms of their left-hand side (to allow for pattern matching)
 - right-hand sides *may not* contain data symbols otherwise

$$\begin{aligned}\text{Last}(h; []) &\rightarrow h \\ \text{Last}(h_1; h_2; tl) &\rightarrow \text{Last}(h_2; tl)\end{aligned}$$

Cons-free Term Rewriting:

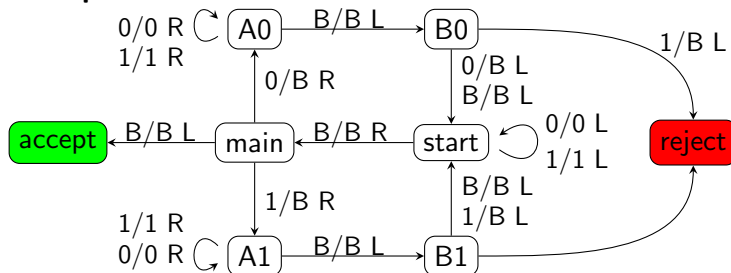
- symbols are split into **data** and **function** symbols
- right-hand sides create no new **data**:
 - right-hand sides *may* contain terms exclusively built of constructors (e.g. True, False, S(0))
 - right-hand sides *may* contain subterms of their left-hand side (to allow for pattern matching)
 - right-hand sides *may not* contain data symbols otherwise

$$\begin{aligned}\text{Last}(h; []) &\rightarrow h \\ \text{Last}(h_1; h_2; tl) &\rightarrow \text{Last}(h_2; tl)\end{aligned}$$

Claim: we can simulate any PTIME Turing Machine

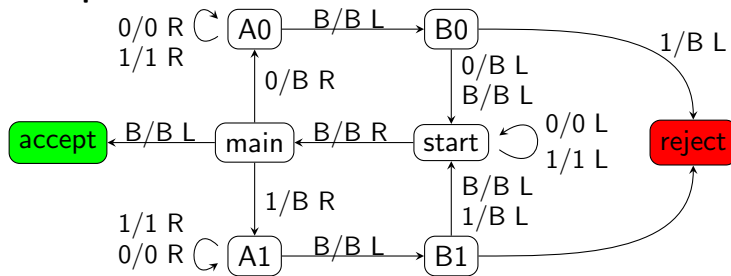
Claim: we can simulate any PTIME Turing Machine

Example:



Claim: we can simulate any PTIME Turing Machine

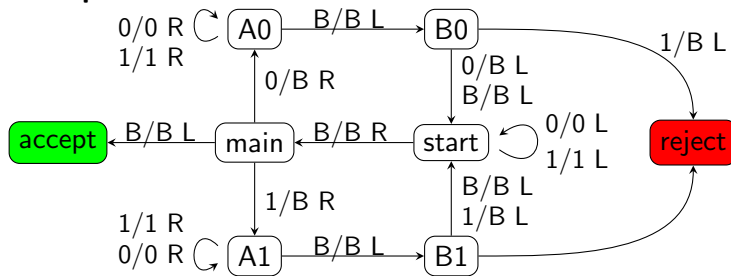
Example:



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

Claim: we can simulate any PTIME Turing Machine

Example:

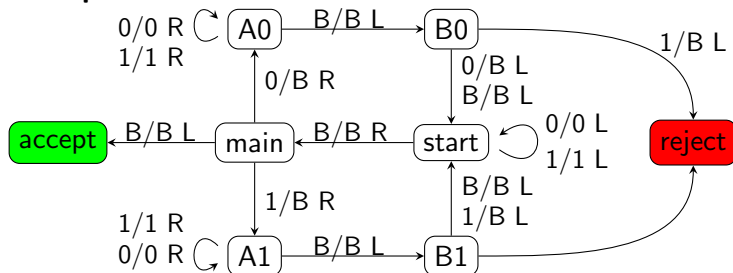


Runs in: $< 2 \cdot (n + 1)^2$ steps

Transition(Start, 0) \rightarrow X(Start, 0, L)

Claim: we can simulate any PTIME Turing Machine

Example:

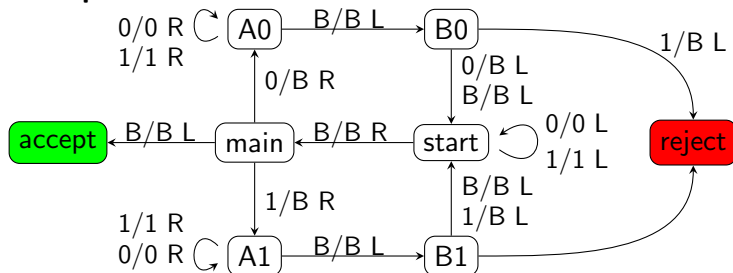


Runs in: $< 2 \cdot (n + 1)^2$ steps

Transition(Start, 0)	→	X(Start, 0, L)
Transition(Start, 1)	→	X(Start, 1, L)
Transition(Start, B)	→	X(Main, B, R)

Claim: we can simulate any PTIME Turing Machine

Example:

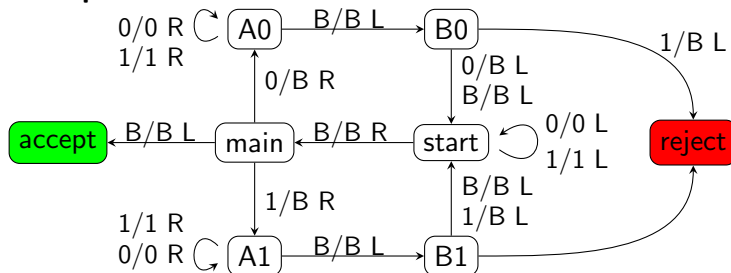


Runs in: $< 2 \cdot (n + 1)^2$ steps

Transition(Start, 0)	→	X(Start, 0, L)
Transition(Start, 1)	→	X(Start, 1, L)
Transition(Start, B)	→	X(Main, B, R)
Transition(Main, 0)	→	X(A0, B, R)

Claim: we can simulate any PTIME Turing Machine

Example:

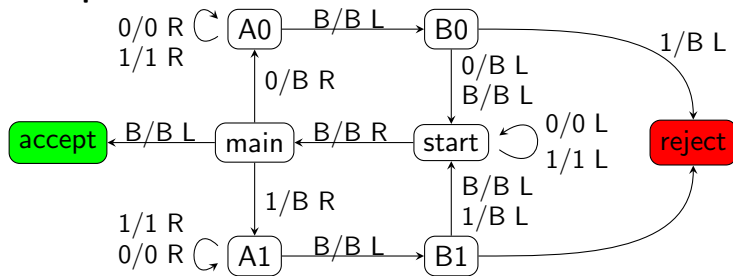


Runs in: $< 2 \cdot (n + 1)^2$ steps

Transition(Start, 0)	\rightarrow	X(Start, 0, L)
Transition(Start, 1)	\rightarrow	X(Start, 1, L)
Transition(Start, B)	\rightarrow	X(Main, B, R)
Transition(Main, 0)	\rightarrow	X(A0, B, R)
Transition(Accept, x)	\rightarrow	X(Accept, x, N)

Claim: we can simulate any PTIME Turing Machine

Example:



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

Transition(Start, 0) → X(Start, 0, L)
 Transition(Start, 1) → X(Start, 1, L)
 Transition(Start, B) → X(Main, B, R)
 Transition(Main, 0) → X(A0, B, R)
 Transition(Accept, x) → X(Accept, x, N)

...

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

Transition(Start, 0) \rightarrow X(Start, 0, L)

Transition(Start, 1) \rightarrow X(Start, 1, L)

Transition(Start, B) \rightarrow X(Main, B, R)

Transition(Main, 0) \rightarrow X(A0, B, R)

Transition(Accept, x) \rightarrow X(Accept, x , S)

...

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

Transition(Start, 0) \rightarrow X(Start, 0, L)

Transition(Start, 1) \rightarrow X(Start, 1, L)

Transition(Start, B) \rightarrow X(Main, B, R)

Transition(Main, 0) \rightarrow X(A0, B, R)

Transition(Accept, x) \rightarrow X(Accept, x , S)

...

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

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

Transition(Start, 0) \rightarrow X(Start, 0, L)

Transition(Start, 1) \rightarrow X(Start, 1, L)

Transition(Start, B) \rightarrow X(Main, B, R)

Transition(Main, 0) \rightarrow X(A0, B, R)

Transition(Accept, x) \rightarrow X(Accept, x , S)

...

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

TransitionAt(i, t_1, t_2, t_3) \rightarrow

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

Transition(Start, 0) \rightarrow X(Start, 0, L)

Transition(Start, 1) \rightarrow X(Start, 1, L)

Transition(Start, B) \rightarrow X(Main, B, R)

Transition(Main, 0) \rightarrow X(A0, B, R)

Transition(Accept, x) \rightarrow X(Accept, x , S)

...

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

TransitionAt(i, t_1, t_2, t_3) \rightarrow Transition(State(i, t_1, t_2, t_3),
CurTape(i, t_1, t_2, t_3))

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

Transition(Start, 0) \rightarrow X(Start, 0, L)

Transition(Start, 1) \rightarrow X(Start, 1, L)

Transition(Start, B) \rightarrow X(Main, B, R)

Transition(Main, 0) \rightarrow X(A0, B, R)

Transition(Accept, x) \rightarrow X(Accept, x , S)

...

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

TransitionAt(i, t_1, t_2, t_3) \rightarrow Transition(State(i, t_1, t_2, t_3),
CurTape(i, t_1, t_2, t_3))

State($i, \square, \square, \square$) \rightarrow

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

Transition(Start, 0) \rightarrow X(Start, 0, L)

Transition(Start, 1) \rightarrow X(Start, 1, L)

Transition(Start, B) \rightarrow X(Main, B, R)

Transition(Main, 0) \rightarrow X(A0, B, R)

Transition(Accept, x) \rightarrow X(Accept, x , S)

...

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

TransitionAt(i, t_1, t_2, t_3) \rightarrow Transition(State(i, t_1, t_2, t_3),
CurTape(i, t_1, t_2, t_3))

State($i, \square, \square, \square$) \rightarrow Start

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

Transition(Start, 0) \rightarrow X(Start, 0, L)

Transition(Start, 1) \rightarrow X(Start, 1, L)

Transition(Start, B) \rightarrow X(Main, B, R)

Transition(Main, 0) \rightarrow X(A0, B, R)

Transition(Accept, x) \rightarrow X(Accept, x , S)

...

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

TransitionAt(i, t_1, t_2, t_3) \rightarrow Transition(State(i, t_1, t_2, t_3),
CurTape(i, t_1, t_2, t_3))

State($i, [], [], []$) \rightarrow Start

State($i, t_1, t_2, x; y$) \rightarrow

State($i, t_1, x; y, []$) \rightarrow

State($i, x; y, [], []$) \rightarrow

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

Transition(Start, 0) \rightarrow X(Start, 0, L)

Transition(Start, 1) \rightarrow X(Start, 1, L)

Transition(Start, B) \rightarrow X(Main, B, R)

Transition(Main, 0) \rightarrow X(A0, B, R)

Transition(Accept, x) \rightarrow X(Accept, x , S)

...

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

TransitionAt(i, t_1, t_2, t_3) \rightarrow Transition(State(i, t_1, t_2, t_3),
CurTape(i, t_1, t_2, t_3))

State($i, [], [], []$) \rightarrow Start

State($i, t_1, t_2, x; y$) \rightarrow

State($i, t_1, x; y, []$) \rightarrow

State($i, x; y, [], []$) \rightarrow

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

Transition(Start, 0) \rightarrow X(Start, 0, L)

Transition(Start, 1) \rightarrow X(Start, 1, L)

Transition(Start, B) \rightarrow X(Main, B, R)

Transition(Main, 0) \rightarrow X(A0, B, R)

Transition(Accept, x) \rightarrow X(Accept, x , S)

...

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

TransitionAt(i, t_1, t_2, t_3) \rightarrow Transition(State(i, t_1, t_2, t_3),
CurTape(i, t_1, t_2, t_3))

State($i, [], [], []$) \rightarrow Start

State($i, t_1, t_2, x; y$) \rightarrow Fst(TransitionAt(i, t_1, t_2, y))

State($i, t_1, x; y, []$) \rightarrow Fst(TransitionAt(i, t_1, y, i))

State($i, x; y, [], []$) \rightarrow Fst(TransitionAt(i, y, i, i))

Conclusion:

Conclusion: any algorithm in PTIME can be implemented by a cons-free TRS

Conclusion: any algorithm in PTIME can be implemented by a cons-free TRS

Recall: cons-free programs with data order 0 [characterise](#) PTIME

Conclusion: any algorithm in PTIME can be implemented by a cons-free TRS

Recall: cons-free programs with data order 0 [characterise](#) PTIME

Question:

does (first-order) term rewriting
characterise PTIME?

Conclusion: any algorithm in PTIME can be implemented by a cons-free TRS

Recall: cons-free programs with data order 0 [characterise](#) PTIME

Question:

does (first-order) term rewriting
characterise PTIME?

Answer:

NO

Conclusion: any algorithm in PTIME can be implemented by a **deterministic** cons-free TRS **with call-by-value reduction**

Recall: cons-free programs with data order 0 **characterise** PTIME

Question:

does (first-order) term rewriting
characterise PTIME?

Answer:

NO

Conclusion: any algorithm in PTIME can be implemented by a **deterministic** cons-free TRS **with call-by-value reduction**

Recall: cons-free programs with data order 0 **characterise** PTIME

Question:

does (first-order) **deterministic** term rewriting
characterise PTIME?

Conclusion: any algorithm in PTIME can be implemented by a **deterministic** cons-free TRS **with call-by-value reduction**

Recall: cons-free programs with data order 0 **characterise** PTIME

Question:

does (first-order) deterministic term rewriting
characterise PTIME?

Answer:

YES

Conclusion: any algorithm in PTIME can be implemented by a **deterministic** cons-free TRS **with call-by-value reduction**

Recall: cons-free programs with data order 0 **characterise** PTIME

Question:

does (first-order) **call-by-value** term rewriting
characterise PTIME?

Answer:

Conclusion: any algorithm in PTIME can be implemented by a **deterministic** cons-free TRS **with call-by-value reduction**

Recall: cons-free programs with data order 0 **characterise** PTIME

Question:

does (first-order) call-by-value term rewriting
characterise PTIME?

Answer:

YES

Overview

- 1 Motivation
- 2 First-order Term Rewriting
- 3 Characterising PTIME
- 4 Call-by-value Rewriting**
- 5 Other Restrictions
- 6 Conclusions

Call-by-value term rewriting characterises PTIME

Call-by-value term rewriting characterises PTIME

- any algorithm in PTIME can be implemented by a cons-free TRS with call-by-value reduction

Call-by-value term rewriting characterises PTIME

- any algorithm in PTIME can be implemented by a cons-free TRS with call-by-value reduction
- for any cons-free TRS with call-by-value reduction, there is an algorithm in PTIME which, given s_1, \dots, s_n , determines whether $\text{Start}(s_1, \dots, s_n) \rightarrow^* \text{True}$

Call-by-value term rewriting characterises PTIME

- any algorithm in PTIME can be implemented by a cons-free TRS with call-by-value reduction
- for any cons-free TRS with call-by-value reduction, there is an algorithm in PTIME which, given s_1, \dots, s_n , determines whether $\text{Start}(s_1, \dots, s_n) \rightarrow^* \text{True}$

Constructing an algorithm

Constructing an algorithm

Rules:

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

Constructing an algorithm

Rules:

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

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

Constructing an algorithm

Rules:

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

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

Constructing an algorithm

Rules:

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

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

Let $\mathcal{B} =$

Constructing an algorithm

Rules:

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

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

Let $\mathcal{B} =$

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

Constructing an algorithm

Rules:

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

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

Let $\mathcal{B} =$

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

Constructing an algorithm

Rules:

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

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

Let $\mathcal{B} =$

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

Constructing an algorithm

Rules:

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

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

Let $\mathcal{B} =$

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

Constructing an algorithm

Rules:

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

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

Let $\mathcal{B} =$

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

Constructing an algorithm

Rules:

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

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

Let $\mathcal{B} =$

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

Constructing an algorithm

Rules:

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

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

Let $\mathcal{B} =$

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

Constructing an algorithm

Rules:

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

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

Let $\mathcal{B} =$

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

Constructing an algorithm

Rules:

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

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

Let $\mathcal{B} =$

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

Constructing an algorithm

Rules:

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

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

Let $\mathcal{B} =$

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

Constructing an algorithm

Rules:

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

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

Let $\mathcal{B} =$

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

Constructing an algorithm

Rules:

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

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

Let $\mathcal{B} =$

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

Make a list:

Constructing an algorithm

Rules:

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

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

Let $\mathcal{B} =$

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

Make a list:

$$\text{Subseq}(0; 0; 1; [], 0; 0; 1; []) \rightarrow^* \{\}$$

Constructing an algorithm

Rules:

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

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

Let $\mathcal{B} =$

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

Make a list:

$$\begin{array}{ll}
 \text{Subseq}(0; 0; 1; [], 0; 0; 1; []) & \rightarrow^* \{\} \\
 \text{Subseq}(0; 0; 1; [], 0; 1; []) & \rightarrow^* \{\}
 \end{array}$$

Constructing an algorithm

Rules:

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

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

Let $\mathcal{B} =$

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

Make a list:

$$\begin{array}{ll}
 \text{Subseq}(0; 0; 1; [], 0; 0; 1; []) & \rightarrow^* \{\} \\
 \text{Subseq}(0; 0; 1; [], 0; 1; []) & \rightarrow^* \{\} \\
 \dots &
 \end{array}$$

Constructing an algorithm

Rules:

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

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

Let $\mathcal{B} =$

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

Make a list:

$$\begin{array}{ll}
 \text{Subseq}(0; 0; 1; [], 0; 0; 1; []) \rightarrow^* \{\} \\
 \text{Subseq}(0; 0; 1; [], 0; 1; []) \rightarrow^* \{\} \\
 \dots \\
 \text{Subseq}(0; 1; [], 0; 0; 1; []) \rightarrow^* \{\}
 \end{array}$$

Constructing an algorithm

Rules:

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

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

Let $\mathcal{B} =$

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

Make a list:

$$\begin{array}{ll}
 \text{Subseq}(0; 0; 1; [], 0; 0; 1; []) \rightarrow^* \{\} \\
 \text{Subseq}(0; 0; 1; [], 0; 1; []) \rightarrow^* \{\} \\
 \dots \\
 \text{Subseq}(0; 1; [], 0; 0; 1; []) \rightarrow^* \{\} \\
 \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* \\
 \dots & \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^* \\
 \dots &
 \end{array}$$

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

Statements:

$$\text{tl}(0; 1; 0; 1; []) \rightarrow^*$$

$$\text{tl}(1; 0; 1; []) \rightarrow^*$$

...

$$\text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 0; 1; [], 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 0; 1; [], []) \rightarrow^*$$

$$\text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 1; [], 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 1; [], 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(1; [], 1; []) \rightarrow^*$$

$$\text{Subseq}([], []) \rightarrow^*$$

...

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* \\
 \dots & \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^* \\
 \dots &
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^*
 \end{array}$$

...

$$\text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 0; 1; [], 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 0; 1; [], []) \rightarrow^*$$

$$\text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 1; [], 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 1; [], 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(1; [], 1; []) \rightarrow^*$$

$$\text{Subseq}([], []) \rightarrow^*$$

...

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* \\
 \dots & \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^* \\
 \dots &
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* \\
 \dots & \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^* \\
 \dots &
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 \dots & \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^* \\
 \dots &
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; []
 \end{array}$$

...

$$\begin{array}{ll}
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^*
 \end{array}$$

...

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

Statements:

$$\text{tl}(0; 1; 0; 1; []) \rightarrow^* 1; 0; 1; []$$

$$\text{tl}(1; 0; 1; []) \rightarrow^* 0; 1; []$$

...

$$\text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 0; 1; [], 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 0; 1; [], []) \rightarrow^*$$

$$\text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 1; [], 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 1; [], 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(1; [], 1; []) \rightarrow^*$$

$$\text{Subseq}([], []) \rightarrow^*$$

...

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

Statements:

$$\text{tl}(0; 1; 0; 1; []) \rightarrow^* 1; 0; 1; []$$

$$\text{tl}(1; 0; 1; []) \rightarrow^* 0; 1; []$$

...

$$\text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 0; 1; [], 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 0; 1; [], []) \rightarrow^*$$

$$\text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 1; [], 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 1; [], 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(1; [], 1; []) \rightarrow^*$$

$$\text{Subseq}([], []) \rightarrow^*$$

...

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

Statements:

$$\text{tl}(0; 1; 0; 1; []) \rightarrow^* 1; 0; 1; []$$

$$\text{tl}(1; 0; 1; []) \rightarrow^* 0; 1; []$$

...

$$\text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 0; 1; [], 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 0; 1; [], []) \rightarrow^*$$

$$\text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 1; [], 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 1; [], 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(1; [], 1; []) \rightarrow^*$$

$$\text{Subseq}([], []) \rightarrow^*$$

...

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

Statements:

$$\text{tl}(0; 1; 0; 1; []) \rightarrow^* 1; 0; 1; []$$

$$\text{tl}(1; 0; 1; []) \rightarrow^* 0; 1; []$$

...

$$\text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 0; 1; [], 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 0; 1; [], []) \rightarrow^*$$

$$\text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 1; [], 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 1; [], 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(1; [], 1; []) \rightarrow^*$$

$$\text{Subseq}([], []) \rightarrow^*$$

...

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

Statements:

$$\text{tl}(0; 1; 0; 1; []) \rightarrow^* 1; 0; 1; []$$

$$\text{tl}(1; 0; 1; []) \rightarrow^* 0; 1; []$$

...

$$\text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 0; 1; [], 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 0; 1; [], []) \rightarrow^*$$

$$\text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 1; [], 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 1; [], 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(1; [], 1; []) \rightarrow^*$$

$$\text{Subseq}([], []) \rightarrow^*$$

...

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; []
 \end{array}$$

...

$$\begin{array}{ll}
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^*
 \end{array}$$

...

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; []
 \end{array}$$

...

$$\text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 0; 1; [], 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 0; 1; [], []) \rightarrow^*$$

$$\text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 1; [], 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 1; [], 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(1; [], 1; []) \rightarrow^*$$

$$\text{Subseq}([], []) \rightarrow^*$$

...

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

Statements:

$$\text{tl}(0; 1; 0; 1; []) \rightarrow^* 1; 0; 1; []$$

$$\text{tl}(1; 0; 1; []) \rightarrow^* 0; 1; []$$

...

$$\text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 0; 1; [], 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 0; 1; [], []) \rightarrow^*$$

$$\text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 1; [], 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 1; [], 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(1; [], 1; []) \rightarrow^*$$

$$\text{Subseq}([], []) \rightarrow^*$$

...

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

Statements:

$$\text{tl}(0; 1; 0; 1; []) \rightarrow^* 1; 0; 1; []$$

$$\text{tl}(1; 0; 1; []) \rightarrow^* 0; 1; []$$

...

$$\text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 0; 1; [], 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 0; 1; [], []) \rightarrow^*$$

$$\text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 1; [], 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 1; [], 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(1; [], 1; []) \rightarrow^*$$

$$\text{Subseq}([], []) \rightarrow^*$$

...

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^* \\
 & \dots
 \end{array}$$

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

Statements:

$$\text{tl}(0; 1; 0; 1; []) \rightarrow^* 1; 0; 1; []$$

$$\text{tl}(1; 0; 1; []) \rightarrow^* 0; 1; []$$

...

$$\text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 0; 1; [], 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 0; 1; [], []) \rightarrow^*$$

$$\text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 1; [], 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 1; [], 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(1; [], 1; []) \rightarrow^*$$

$$\text{Subseq}([], []) \rightarrow^*$$

...

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^* \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^* \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^* \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^* \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^* \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^* \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^* \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^* \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^* \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^* \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^* \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^* \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 \dots & \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 \dots &
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\text{tl}(0; 1; 0; 1; []) \rightarrow^* 1; 0; 1; []$$

$$\text{tl}(1; 0; 1; []) \rightarrow^* 0; 1; []$$

...

$$\text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 0; 1; [], 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 0; 1; [], []) \rightarrow^* \text{False}$$

$$\text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 1; [], 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 1; [], 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(1; [], 1; []) \rightarrow^*$$

$$\text{Subseq}([], []) \rightarrow^* \text{True, False}$$

...

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 \dots & \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 \dots &
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) \rightarrow^* 1; 0; 1; [] & \\
 \text{tl}(1; 0; 1; []) \rightarrow^* 0; 1; [] & \\
 \dots & \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^* & \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) \rightarrow^* & \\
 \text{Subseq}(0; 0; 1; [], []) \rightarrow^* \text{False} & \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^* & \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) \rightarrow^* & \\
 \text{Subseq}(0; 1; [], 0; 1; []) \rightarrow^* & \\
 \text{Subseq}(1; [], 1; []) \rightarrow^* \text{True, False} & \\
 \text{Subseq}([], []) \rightarrow^* \text{True, False} & \\
 \dots &
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) \rightarrow^* 1; 0; 1; [] & \\
 \text{tl}(1; 0; 1; []) \rightarrow^* 0; 1; [] & \\
 \dots & \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^* & \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) \rightarrow^* & \\
 \text{Subseq}(0; 0; 1; [], []) \rightarrow^* \text{False} & \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^* & \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) \rightarrow^* & \\
 \text{Subseq}(0; 1; [], 0; 1; []) \rightarrow^* & \\
 \text{Subseq}(1; [], 1; []) \rightarrow^* \text{True, False} & \\
 \text{Subseq}([], []) \rightarrow^* \text{True, False} & \\
 \dots &
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 \dots & \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 \dots &
 \end{array}$$

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

Statements:

$$\text{tl}(0; 1; 0; 1; []) \rightarrow^* 1; 0; 1; []$$

$$\text{tl}(1; 0; 1; []) \rightarrow^* 0; 1; []$$

...

$$\text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 0; 1; [], 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 0; 1; [], []) \rightarrow^* \text{False}$$

$$\text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 1; [], 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 1; [], 0; 1; []) \rightarrow^* \text{True, False}$$

$$\text{Subseq}(1; [], 1; []) \rightarrow^* \text{True, False}$$

$$\text{Subseq}([], []) \rightarrow^* \text{True, False}$$

...

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 \dots & \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 \dots &
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\text{tl}(0; 1; 0; 1; []) \rightarrow^* 1; 0; 1; []$$

$$\text{tl}(1; 0; 1; []) \rightarrow^* 0; 1; []$$

...

$$\text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^* \text{True, False}$$

$$\text{Subseq}(0; 0; 1; [], 1; 0; 1; []) \rightarrow^* \text{False}$$

$$\text{Subseq}(0; 0; 1; [], []) \rightarrow^* \text{False}$$

$$\text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^*$$

$$\text{Subseq}(0; 1; [], 1; 0; 1; []) \rightarrow^* \text{True, False}$$

$$\text{Subseq}(0; 1; [], 0; 1; []) \rightarrow^* \text{True, False}$$

$$\text{Subseq}(1; [], 1; []) \rightarrow^* \text{True, False}$$

$$\text{Subseq}([], []) \rightarrow^* \text{True, False}$$

...

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 \dots & \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 \dots &
 \end{array}$$

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

Statements:

$$\text{tl}(0; 1; 0; 1; []) \rightarrow^* 1; 0; 1; []$$

$$\text{tl}(1; 0; 1; []) \rightarrow^* 0; 1; []$$

...

$$\text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^* \text{True, False}$$

$$\text{Subseq}(0; 0; 1; [], 1; 0; 1; []) \rightarrow^* \text{False}$$

$$\text{Subseq}(0; 0; 1; [], []) \rightarrow^* \text{False}$$

$$\text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^* \text{True, False}$$

$$\text{Subseq}(0; 1; [], 1; 0; 1; []) \rightarrow^* \text{True, False}$$

$$\text{Subseq}(0; 1; [], 0; 1; []) \rightarrow^* \text{True, False}$$

$$\text{Subseq}(1; [], 1; []) \rightarrow^* \text{True, False}$$

$$\text{Subseq}([], []) \rightarrow^* \text{True, False}$$

...

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

Statements:

$$\text{tl}(0; 1; 0; 1; []) \rightarrow^* 1; 0; 1; []$$

$$\text{tl}(1; 0; 1; []) \rightarrow^* 0; 1; []$$

...

$$\text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^* \text{True, False}$$

$$\text{Subseq}(0; 0; 1; [], 1; 0; 1; []) \rightarrow^* \text{False}$$

$$\text{Subseq}(0; 0; 1; [], []) \rightarrow^* \text{False}$$

$$\text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^* \text{True, False}$$

$$\text{Subseq}(0; 1; [], 1; 0; 1; []) \rightarrow^* \text{True, False}$$

$$\text{Subseq}(0; 1; [], 0; 1; []) \rightarrow^* \text{True, False}$$

$$\text{Subseq}(1; [], 1; []) \rightarrow^* \text{True, False}$$

$$\text{Subseq}([], []) \rightarrow^* \text{True, False}$$

...

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\begin{array}{ll}
 \text{tl}(0; 1; 0; 1; []) & \rightarrow^* 1; 0; 1; [] \\
 \text{tl}(1; 0; 1; []) & \rightarrow^* 0; 1; [] \\
 & \dots \\
 \text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{False} \\
 \text{Subseq}(0; 0; 1; [], []) & \rightarrow^* \text{False} \\
 \text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 1; 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(0; 1; [], 0; 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}(1; [], 1; []) & \rightarrow^* \text{True, False} \\
 \text{Subseq}([], []) & \rightarrow^* \text{True, False} \\
 & \dots
 \end{array}$$

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

Statements:

$$\text{tl}(0; 1; 0; 1; []) \rightarrow^* 1; 0; 1; []$$

$$\text{tl}(1; 0; 1; []) \rightarrow^* 0; 1; []$$

...

$$\text{Subseq}(0; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^* \text{True, False}$$

$$\text{Subseq}(0; 0; 1; [], 1; 0; 1; []) \rightarrow^* \text{False}$$

$$\text{Subseq}(0; 0; 1; [], []) \rightarrow^* \text{False}$$

$$\text{Subseq}(1; 0; 1; [], 0; 1; 0; 1; []) \rightarrow^* \text{True, False}$$

$$\text{Subseq}(0; 1; [], 1; 0; 1; []) \rightarrow^* \text{True, False}$$

$$\text{Subseq}(0; 1; [], 0; 1; []) \rightarrow^* \text{True, False}$$

$$\text{Subseq}(1; [], 1; []) \rightarrow^* \text{True, False}$$

$$\text{Subseq}([], []) \rightarrow^* \text{True, False}$$

...

Algorithm: formal description

Algorithm: formal description

Given: TRS with rules \mathcal{R}

Algorithm: formal description

Given: TRS with rules \mathcal{R}

Input: start term $\text{Start}(s_1, \dots, s_n)$ with s_1, \dots, s_n data

Algorithm: formal description

Given: TRS with rules \mathcal{R}

Input: start term $\text{Start}(s_1, \dots, s_n)$ with s_1, \dots, s_n data

Intended output: true if $\text{Start}(s_1, \dots, s_n) \rightarrow_{\mathcal{R}}^*$ True or false if not

Algorithm: formal description

Given: TRS with rules \mathcal{R}

Input: start term $\text{Start}(s_1, \dots, s_n)$ with s_1, \dots, s_n data

Intended output: true if $\text{Start}(s_1, \dots, s_n) \rightarrow_{\mathcal{R}}^*$ True or false if not

Initialisation:

- let \mathcal{B} contain all subterms of s and data subterms of rule right-hand sides

Algorithm: formal description

Given: TRS with rules \mathcal{R}

Input: start term $\text{Start}(s_1, \dots, s_n)$ with s_1, \dots, s_n data

Intended output: true if $\text{Start}(s_1, \dots, s_n) \rightarrow_{\mathcal{R}}^*$ True or false if not

Initialisation:

- let \mathcal{B} contain all subterms of s and data subterms of rule right-hand sides
- make a list of all statements $f(t_1, \dots, t_m) \rightarrow^* \{\}$ with $\vec{t} \in \mathcal{B}$

Algorithm: formal description

Given: TRS with rules \mathcal{R}

Input: start term $\text{Start}(s_1, \dots, s_n)$ with s_1, \dots, s_n data

Intended output: true if $\text{Start}(s_1, \dots, s_n) \rightarrow_{\mathcal{R}}^*$ True or false if not

Initialisation:

- let \mathcal{B} contain all subterms of s and data subterms of rule right-hand sides
- make a list of all statements $f(t_1, \dots, t_m) \rightarrow^* \{\}$ with $\vec{t} \in \mathcal{B}$

Repeat:

- find all u such that $f(t_1, \dots, t_m) \rightarrow_{\mathcal{R}} u$

Algorithm: formal description

Given: TRS with rules \mathcal{R}

Input: start term $\text{Start}(s_1, \dots, s_n)$ with s_1, \dots, s_n data

Intended output: true if $\text{Start}(s_1, \dots, s_n) \rightarrow_{\mathcal{R}}^*$ True or false if not

Initialisation:

- let \mathcal{B} contain all subterms of s and data subterms of rule right-hand sides
- make a list of all statements $f(t_1, \dots, t_m) \rightarrow^* \{\}$ with $\vec{t} \in \mathcal{B}$

Repeat:

- find all u such that $f(t_1, \dots, t_m) \rightarrow_{\mathcal{R}} u$
- simplify u using “confirmed steps”

Algorithm: formal description

Given: TRS with rules \mathcal{R}

Input: start term $\text{Start}(s_1, \dots, s_n)$ with s_1, \dots, s_n data

Intended output: true if $\text{Start}(s_1, \dots, s_n) \rightarrow_{\mathcal{R}}^*$ True or false if not

Initialisation:

- let \mathcal{B} contain all subterms of s and data subterms of rule right-hand sides
- make a list of all statements $f(t_1, \dots, t_m) \rightarrow^* \{ \}$ with $\vec{t} \in \mathcal{B}$

Repeat:

- find all u such that $f(t_1, \dots, t_m) \rightarrow_{\mathcal{R}} u$
- simplify u using “confirmed steps”
- if u simplifies to data q , include q in the set

Algorithm: formal description

Given: TRS with rules \mathcal{R}

Input: start term $\text{Start}(s_1, \dots, s_n)$ with s_1, \dots, s_n data

Intended output: true if $\text{Start}(s_1, \dots, s_n) \rightarrow_{\mathcal{R}}^*$ True or false if not

Initialisation:

- let \mathcal{B} contain all subterms of s and data subterms of rule right-hand sides
- make a list of all statements $f(t_1, \dots, t_m) \rightarrow^* \{\}$ with $\vec{t} \in \mathcal{B}$

Repeat:

- find all u such that $f(t_1, \dots, t_m) \rightarrow_{\mathcal{R}} u$
- simplify u using “confirmed steps”
- if u simplifies to data q , include q in the set

Conclusion:

- stop when no new elements are added to sets

Algorithm: formal description

Given: TRS with rules \mathcal{R}

Input: start term $\text{Start}(s_1, \dots, s_n)$ with s_1, \dots, s_n data

Intended output: true if $\text{Start}(s_1, \dots, s_n) \rightarrow_{\mathcal{R}}^*$ True or false if not

Initialisation:

- let \mathcal{B} contain all subterms of s and data subterms of rule right-hand sides
- make a list of all statements $f(t_1, \dots, t_m) \rightarrow^* \{\}$ with $\vec{t} \in \mathcal{B}$

Repeat:

- find all u such that $f(t_1, \dots, t_m) \rightarrow_{\mathcal{R}} u$
- simplify u using “confirmed steps”
- if u simplifies to data q , include q in the set

Conclusion:

- stop when no new elements are added to sets
- return true if and only if $\text{Start}(s_1, \dots, s_n) \rightarrow^* A \cup \{\text{True}\}$

Conclusion:

Conclusion:

- call-by-value term rewriting characterises PTIME

Conclusion:

- call-by-value term rewriting characterises PTIME
- innermost term rewriting characterises PTIME

Conclusion:

- call-by-value term rewriting characterises PTIME
- innermost term rewriting characterises PTIME

Note: non-determinism does not increase expressivity

Conclusion:

- call-by-value term rewriting characterises PTIME
- innermost term rewriting characterises PTIME

Note: non-determinism does not increase expressivity
(see also: Cook '71, Bonfante '06)

Overview

- 1 Motivation
- 2 First-order Term Rewriting
- 3 Characterising PTIME
- 4 Call-by-value Rewriting
- 5 Other Restrictions**
- 6 Conclusions

Recall:

Recall: any algorithm in PTIME can be implemented by a deterministic cons-free TRS with call-by-value reduction

Recall: any algorithm in PTIME can be implemented by a **deterministic** cons-free TRS **with call-by-value reduction**

Question:

does **deterministic** term rewriting characterise PTIME?

Recall: any algorithm in PTIME can be implemented by a **deterministic** cons-free TRS **with call-by-value reduction**

Question:

does **deterministic** term rewriting characterise PTIME?

Answer:

YES

Recall: any algorithm in PTIME can be implemented by a **deterministic** cons-free TRS **with call-by-value reduction**

Question:

does **deterministic** term rewriting characterise PTIME?

Answer:

YES

Proof:

Recall: any algorithm in PTIME can be implemented by a **deterministic** cons-free TRS **with call-by-value reduction**

Question:

does **deterministic** term rewriting characterise PTIME?

Answer:

YES

Proof:

suppose $\rightarrow_{\mathcal{R}}^*$ has unique normal forms

Recall: any algorithm in PTIME can be implemented by a **deterministic** cons-free TRS **with call-by-value reduction**

Question:

does **deterministic** term rewriting characterise PTIME?

Answer:

YES

Proof:

suppose $\rightarrow_{\mathcal{R}}^*$ has unique normal forms
then

Recall: any algorithm in PTIME can be implemented by a **deterministic** cons-free TRS **with call-by-value reduction**

Question:

does **deterministic** term rewriting characterise PTIME?

Answer:

YES

Proof:

suppose $\rightarrow_{\mathcal{R}}^*$ has unique normal forms
then

$s \rightarrow_{\mathcal{R}}^* t$ with t data if and only if $s \rightarrow_{\mathcal{R}, \text{call-by-value}}^* t$

Recall: any algorithm in PTIME can be implemented by a **deterministic** cons-free TRS with **call-by-value reduction**

Question:

does **deterministic** term rewriting characterise PTIME?

Answer:

YES

Proof:

suppose $\rightarrow_{\mathcal{R}}^*$ has unique normal forms

then **exists \mathcal{R}' such that**

$s \rightarrow_{\mathcal{R}}^* t$ with t data if and only if $s \rightarrow_{\mathcal{R}', \text{call-by-value}}^* t$

Recall: any algorithm in PTIME can be implemented by a **deterministic** cons-free TRS with **call-by-value reduction**

Question:

does **deterministic** term rewriting characterise PTIME?

Answer:

YES

Proof:

suppose $\rightarrow_{\mathcal{R}}^*$ has unique normal forms
 then **exists \mathcal{R}' such that**
 $s \rightarrow_{\mathcal{R}}^* t$ with t data if and only if $s \rightarrow_{\mathcal{R}', \text{call-by-value}}^* t$
(in a cons-free TRS)

Semi-Linear Term Rewriting

(adapted from [Carvalho and Simonsen '14])

Semi-Linear Term Rewriting

(adapted from [Carvalho and Simonsen '14])

Definition:

a rule $f(\ell_1, \dots, \ell_n) \rightarrow r$ is **semi-linear** if:

Semi-Linear Term Rewriting

(adapted from [Carvalho and Simonsen '14])

Definition:

a rule $f(\ell_1, \dots, \ell_n) \rightarrow r$ is **semi-linear** if:
for each ℓ_i :

Semi-Linear Term Rewriting

(adapted from [Carvalho and Simonsen '14])

Definition:

a rule $f(\ell_1, \dots, \ell_n) \rightarrow r$ is **semi-linear** if:
for each ℓ_i :

if ℓ_i is a variable then ℓ_i occurs at most once in r

Semi-Linear Term Rewriting

(adapted from [Carvalho and Simonsen '14])

Definition:

a rule $f(\ell_1, \dots, \ell_n) \rightarrow r$ is **semi-linear** if:
for each ℓ_i :

if ℓ_i is a variable then ℓ_i occurs at most once in r

Claim:

rewriting with **semi-linear rules** characterises PTIME

Semi-Linear Term Rewriting

(adapted from [Carvalho and Simonsen '14])

Definition:

a rule $f(\ell_1, \dots, \ell_n) \rightarrow r$ is **semi-linear** if:
for each ℓ_i :

if ℓ_i is a variable then ℓ_i occurs at most once in r

Claim:

rewriting with **semi-linear rules** characterises PTIME

Proof:

Semi-Linear Term Rewriting

(adapted from [Carvalho and Simonsen '14])

Definition:

a rule $f(\ell_1, \dots, \ell_n) \rightarrow r$ is **semi-linear** if:
for each ℓ_i :

if ℓ_i is a variable then ℓ_i occurs at most once in r

Claim:

rewriting with **semi-linear rules** characterises PTIME

Proof:

- same algorithm as for innermost can be expressed semi-linear

Semi-Linear Term Rewriting

(adapted from [Carvalho and Simonsen '14])

Definition:

a rule $f(\ell_1, \dots, \ell_n) \rightarrow r$ is **semi-linear** if:
for each ℓ_i :

if ℓ_i is a variable then ℓ_i occurs at most once in r

Claim:

rewriting with **semi-linear rules** characterises PTIME

Proof:

- same algorithm as for innermost can be expressed semi-linear
- if all rules in \mathcal{R} are semi-linear
then exists \mathcal{R}' such that
 $s \rightarrow_{\mathcal{R}}^* t$ with t data if and only if $s \rightarrow_{\mathcal{R}', \text{call-by-value}}^* t$

Definition:

a rule $f(\ell_1, \dots, \ell_n) \rightarrow r$ is **semi-linear** if:

for each ℓ_i :

if ℓ_i is a variable then ℓ_i occurs at most once in r

Definition:

a rule $f(\ell_1, \dots, \ell_n) \rightarrow r$ is **semi-linear** if:

for each ℓ_i :

if ℓ_i is a variable then ℓ_i occurs at most once in r

What does this prohibit?

Definition:

a rule $f(\ell_1, \dots, \ell_n) \rightarrow r$ is **semi-linear** if:

for each ℓ_i :

if ℓ_i is a variable then ℓ_i occurs at most once in r

What does this prohibit?

$\text{either}(x, y) \rightarrow x$

$\text{either}(x, y) \rightarrow y$

Definition:

a rule $f(\ell_1, \dots, \ell_n) \rightarrow r$ is **semi-linear** if:

for each ℓ_i :

if ℓ_i is a variable then ℓ_i occurs at most once in r

What does this prohibit?

either(x, y) $\rightarrow x$

either(x, y) $\rightarrow y$

f(x, n) \rightarrow ifelse(equal(x, n), g(x), g($x, \text{pred}(n)$))

Definition:

a rule $f(\ell_1, \dots, \ell_n) \rightarrow r$ is **semi-linear** if:

for each ℓ_i :

if ℓ_i is a variable then ℓ_i occurs at most once in r

What does this prohibit?

either(x, y) $\rightarrow x$

either(x, y) $\rightarrow y$

$f(x, n) \rightarrow \text{ifelse}(\text{equal}(x, n), g(x), g(x, \text{pred}(n)))$

$f(\text{either}(0, \text{either}(2)), 1)$

Definition:

a rule $f(\ell_1, \dots, \ell_n) \rightarrow r$ is **semi-linear** if:

for each ℓ_i :

if ℓ_i is a variable then ℓ_i occurs at most once in r

What does this prohibit?

either(x, y) $\rightarrow x$

either(x, y) $\rightarrow y$

$f(x, n) \rightarrow \text{ifelse}(\text{equal}(x, n), g(x), g(x, \text{pred}(n)))$

$f(\text{either}(0, \text{either}(2)), 1)$

$\rightarrow \text{ifelse}(\text{equal}(\text{either}(0, \text{either}(2)), 1), g(\dots), f(\dots, \text{pred}(1)))$

Definition:

a rule $f(\ell_1, \dots, \ell_n) \rightarrow r$ is **semi-linear** if:

for each ℓ_i :

if ℓ_i is a variable then ℓ_i occurs at most once in r

What does this prohibit?

either(x, y) $\rightarrow x$

either(x, y) $\rightarrow y$

$f(x, n) \rightarrow \text{ifelse}(\text{equal}(x, n), g(x), g(x, \text{pred}(n)))$

$f(\text{either}(0, \text{either}(2)), 1)$

$\rightarrow \text{ifelse}(\text{equal}(\text{either}(0, \text{either}(2)), 1), g(\dots), f(\dots, \text{pred}(1)))$

$\rightarrow \text{ifelse}(\text{equal}(2, 1), g(\dots), f(\dots, \text{pred}(1)))$

Definition:

a rule $f(\ell_1, \dots, \ell_n) \rightarrow r$ is **semi-linear** if:

for each ℓ_i :

if ℓ_i is a variable then ℓ_i occurs at most once in r

What does this prohibit?

either(x, y) $\rightarrow x$

either(x, y) $\rightarrow y$

$f(x, n) \rightarrow \text{ifelse}(\text{equal}(x, n), g(x), g(x, \text{pred}(n)))$

$f(\text{either}(0, \text{either}(2)), 1)$

$\rightarrow \text{ifelse}(\text{equal}(\text{either}(0, \text{either}(2)), 1), g(\dots), f(\dots, \text{pred}(1)))$

$\rightarrow \text{ifelse}(\text{equal}(2, 1), g(\dots), f(\dots, \text{pred}(1)))$

$\rightarrow^* f(\text{either}(0, \text{either}(2)), \text{pred}(1))$

Definition:

a rule $f(\ell_1, \dots, \ell_n) \rightarrow r$ is **semi-linear** if:

for each ℓ_i :

if ℓ_i is a variable then ℓ_i occurs at most once in r

What does this prohibit?

either(x, y) $\rightarrow x$

either(x, y) $\rightarrow y$

$f(x, n) \rightarrow \text{ifelse}(\text{equal}(x, n), g(x), g(x, \text{pred}(n)))$

$f(\text{either}(0, \text{either}(2)), 1)$

$\rightarrow \text{ifelse}(\text{equal}(\text{either}(0, \text{either}(2)), 1), g(\dots), f(\dots, \text{pred}(1)))$

$\rightarrow \text{ifelse}(\text{equal}(2, 1), g(\dots), f(\dots, \text{pred}(1)))$

$\rightarrow^* f(\text{either}(0, \text{either}(2)), \text{pred}(1))$

$\rightarrow f(\text{either}(0, \text{either}(2)), 0)$

Definition:

a rule $f(\ell_1, \dots, \ell_n) \rightarrow r$ is **semi-linear** if:

for each ℓ_i :

if ℓ_i is a variable then ℓ_i occurs at most once in r

What does this prohibit?

either(x, y) $\rightarrow x$

either(x, y) $\rightarrow y$

$f(x, n) \rightarrow \text{ifelse}(\text{equal}(x, n), g(x), g(x, \text{pred}(n)))$

$f(\text{either}(0, \text{either}(2)), 1)$

$\rightarrow \text{ifelse}(\text{equal}(\text{either}(0, \text{either}(2)), 1), g(\dots), f(\dots, \text{pred}(1)))$

$\rightarrow \text{ifelse}(\text{equal}(2, 1), g(\dots), f(\dots, \text{pred}(1)))$

$\rightarrow^* f(\text{either}(0, \text{either}(2)), \text{pred}(1))$

$\rightarrow f(\text{either}(0, \text{either}(2)), 0)$

$\rightarrow \text{ifelse}(\text{equal}(\text{either}(0, \text{either}(2)), 0), g(\dots), f(\dots, \text{pred}(0)))$

Definition:

a rule $f(\ell_1, \dots, \ell_n) \rightarrow r$ is **semi-linear** if:

for each ℓ_i :

if ℓ_i is a variable then ℓ_i occurs at most once in r

What does this prohibit?

either(x, y) $\rightarrow x$

either(x, y) $\rightarrow y$

$f(x, n) \rightarrow \text{ifelse}(\text{equal}(x, n), g(x), g(x, \text{pred}(n)))$

$f(\text{either}(0, \text{either}(2)), 1)$

$\rightarrow \text{ifelse}(\text{equal}(\text{either}(0, \text{either}(2)), 1), g(\dots), f(\dots, \text{pred}(1)))$

$\rightarrow \text{ifelse}(\text{equal}(2, 1), g(\dots), f(\dots, \text{pred}(1)))$

$\rightarrow^* f(\text{either}(0, \text{either}(2)), \text{pred}(1))$

$\rightarrow f(\text{either}(0, \text{either}(2)), 0)$

$\rightarrow \text{ifelse}(\text{equal}(\text{either}(0, \text{either}(2)), 0), g(\dots), f(\dots, \text{pred}(0)))$

$\rightarrow \text{ifelse}(\text{equal}(0, 0), g(\dots), f(\dots, \text{pred}(0)))$

Definition:

a rule $f(\ell_1, \dots, \ell_n) \rightarrow r$ is **semi-linear** if:

for each ℓ_i :

if ℓ_i is a variable then ℓ_i occurs at most once in r

What does this prohibit?

either(x, y) $\rightarrow x$

either(x, y) $\rightarrow y$

$f(x, n) \rightarrow \text{ifelse}(\text{equal}(x, n), g(x), g(x, \text{pred}(n)))$

$f(\text{either}(0, \text{either}(2)), 1)$

$\rightarrow \text{ifelse}(\text{equal}(\text{either}(0, \text{either}(2)), 1), g(\dots), f(\dots, \text{pred}(1)))$

$\rightarrow \text{ifelse}(\text{equal}(2, 1), g(\dots), f(\dots, \text{pred}(1)))$

$\rightarrow^* f(\text{either}(0, \text{either}(2)), \text{pred}(1))$

$\rightarrow f(\text{either}(0, \text{either}(2)), 0)$

$\rightarrow \text{ifelse}(\text{equal}(\text{either}(0, \text{either}(2)), 0), g(\dots), f(\dots, \text{pred}(0)))$

$\rightarrow \text{ifelse}(\text{equal}(0, 0), g(\dots), f(\dots, \text{pred}(0)))$

$\rightarrow g(\text{either}(0, \text{either}(2)))$

Definition:

a rule $f(\ell_1, \dots, \ell_n) \rightarrow r$ is **semi-linear** if:

for each ℓ_i :

if ℓ_i is a variable then ℓ_i occurs at most once in r

What does this prohibit?

either(x, y) $\rightarrow x$

either(x, y) $\rightarrow y$

f(x, n) \rightarrow ifelse(equal(x, n), g(x), g(x , pred(n)))

f(either(0, either(2)), 1)

\rightarrow ifelse(equal(either(0, either(2)), 1), g(...), f(..., pred(1)))

\rightarrow ifelse(equal(2, 1), g(...), f(..., pred(1)))

\rightarrow^* f(either(0, either(2)), pred(1))

\rightarrow f(either(0, either(2)), 0)

\rightarrow ifelse(equal(either(0, either(2)), 0), g(...), f(..., pred(0)))

\rightarrow ifelse(equal(0, 0), g(...), f(..., pred(0)))

\rightarrow g(either(0, either(2)))

Definition:

a rule $f(l_1, \dots, l_n) \rightarrow r$ is **semi-linear** if:
for each l_i :

if l_i is a variable then l_i occurs at most once in r

What does this allow?

Definition:

a rule $f(l_1, \dots, l_n) \rightarrow r$ is **semi-linear** if:
for each l_i :

if l_i is a variable then l_i occurs at most once in r

What does this allow?

- duplication of variables **which must be instantiated by irreducible terms**

Definition:

a rule $f(l_1, \dots, l_n) \rightarrow r$ is **semi-linear** if:
for each l_i :

if l_i is a variable then l_i occurs at most once in r

What does this allow?

- duplication of variables **which must be instantiated by irreducible terms** (due to cons-freeness)

Definition:

a rule $f(l_1, \dots, l_n) \rightarrow r$ is **semi-linear** if:
for each l_i :

if l_i is a variable then l_i occurs at most once in r

What does this allow?

- duplication of variables **which must be instantiated by irreducible terms** (due to cons-freeness)
- passing on reducible terms **but never copying**

Definition:

a rule $f(l_1, \dots, l_n) \rightarrow r$ is **semi-linear** if:
for each l_i :

if l_i is a variable then l_i occurs at most once in r

What does this allow?

- duplication of variables **which must be instantiated by irreducible terms** (due to cons-freeness)
- passing on reducible terms **but never copying**
(so in $f(\text{either}(0, \text{either}(1, 2)))$) we cannot *both* test reducts *and* pass on the term)

Definition:

a rule $f(\ell_1, \dots, \ell_n) \rightarrow r$ is **semi-linear** if:
for each ℓ_i :

if ℓ_i is a variable then ℓ_i occurs at most once in r

What does this allow?

- duplication of variables **which must be instantiated by irreducible terms** (due to cons-freeness)
- passing on reducible terms **but never copying**
(so in $f(\text{either}(0, \text{either}(1, 2)))$) we cannot *both* test reducts *and* pass on the term)

\implies we might as well reduce subterms immediately

Definition:

a rule $f(\ell_1, \dots, \ell_n) \rightarrow r$ is **semi-linear** if:
for each ℓ_i :

if ℓ_i is a variable then ℓ_i occurs at most once in r

What does this allow?

- duplication of variables **which must be instantiated by irreducible terms** (due to cons-freeness)
- passing on reducible terms **but never copying**
(so in $f(\text{either}(0, \text{either}(1, 2)))$) we cannot *both* test reducts *and* pass on the term)

\implies we might as well reduce subterms immediately

Caution:

$$\begin{array}{l} f(x) \rightarrow 0 \\ a \rightarrow a \end{array}$$

Definition:

a rule $f(\ell_1, \dots, \ell_n) \rightarrow r$ is **semi-linear** if:
for each ℓ_i :

if ℓ_i is a variable then ℓ_i occurs at most once in r

What does this allow?

- duplication of variables **which must be instantiated by irreducible terms** (due to cons-freeness)
- passing on reducible terms **but never copying**
(so in $f(\text{either}(0, \text{either}(1, 2)))$) we cannot *both* test reducts *and* pass on the term)

\implies we might as well reduce subterms immediately

Caution:

$$\begin{array}{l} f(x) \rightarrow 0 \\ a \rightarrow a \end{array}$$

$$f(a)$$

Semi-Linear Term Rewriting

Definition:

a rule $f(l_1, \dots, l_n) \rightarrow r$ is **semi-linear** if:
for each l_i :

if l_i is a variable then l_i occurs at most once in r

Claim:

rewriting with **semi-linear** rules characterises PTIME

Proof:

- same algorithm as for innermost can be expressed semi-linear
- if all rules in \mathcal{R} are semi-linear
then exists \mathcal{R}' such that
 $s \rightarrow_{\mathcal{R}}^* t$ with t data if and only if $s \rightarrow_{\mathcal{R}', \text{call-by-value}}^* t$

Semi-Linear Term Rewriting

Definition:

a rule $f(\ell_1, \dots, \ell_n) \rightarrow r$ is **semi-linear** if:
for each ℓ_i :

if ℓ_i is a variable then ℓ_i occurs at most once in r

Claim:

rewriting with **semi-linear** rules characterises PTIME

Proof:

- same algorithm as for innermost can be expressed semi-linear
- if all rules in \mathcal{R} are semi-linear
then let $\mathcal{R}' = \mathcal{R} \cup \{f(x_1, \dots, x_n) \rightarrow \perp \mid f \text{ a function symbol}\}$
then $s \rightarrow_{\mathcal{R}}^* t$ with t data if and only if $s \rightarrow_{\mathcal{R}', \text{call-by-value}}^* t$

Semi-Linear Term Rewriting

Definition:

a rule $f(\ell_1, \dots, \ell_n) \rightarrow r$ is **semi-linear** if:
for each ℓ_i :

if ℓ_i is a variable then ℓ_i occurs at most once in r

Claim:

rewriting with **confluent** rules characterises PTIME

Proof:

- same algorithm as for innermost **is confluent**
- if all rules in \mathcal{R} are **confluent**
then let $\mathcal{R}' = \mathcal{R} \cup \{f(x_1, \dots, x_n) \rightarrow \perp \mid f \text{ a function symbol}\}$
then $s \rightarrow_{\mathcal{R}}^* t$ with t data if and only if $s \rightarrow_{\mathcal{R}', \text{call-by-value}}^* t$

“Constrained” Term Rewriting [Carvalho and Simonsen '14]

“Constrained” Term Rewriting [Carvalho and Simonsen '14]

Definition: there is $\mathcal{A} \subseteq \mathcal{F}$ such that:

- all rules $f(\ell_1, \dots, \ell_n) \rightarrow r$ with $f \in \mathcal{A}$ are semi-linear
- for all rules $f(\ell_1, \dots, \ell_n) \rightarrow r$ with $f \notin \mathcal{A}$: if ℓ_i is a variable, and $r = C[g(t_1, \dots, D[\ell_i], \dots, t_m)]$ then $g \in \mathcal{A}$

“Constrained” Term Rewriting [Carvalho and Simonsen '14]

Definition: there is $\mathcal{A} \subseteq \mathcal{F}$ such that:

- all rules $f(\ell_1, \dots, \ell_n) \rightarrow r$ with $f \in \mathcal{A}$ are semi-linear
- for all rules $f(\ell_1, \dots, \ell_n) \rightarrow r$ with $f \notin \mathcal{A}$: if ℓ_i is a variable, and $r = C[g(t_1, \dots, D[\ell_i], \dots, t_m)]$ then $g \in \mathcal{A}$

\implies easily adapted to a system with only semi-linear rules

“Constrained” Term Rewriting [Carvalho and Simonsen '14]

Definition: there is $\mathcal{A} \subseteq \mathcal{F}$ such that:

- all rules $f(\ell_1, \dots, \ell_n) \rightarrow r$ with $f \in \mathcal{A}$ are semi-linear
- for all rules $f(\ell_1, \dots, \ell_n) \rightarrow r$ with $f \notin \mathcal{A}$: if ℓ_i is a variable, and $r = C[g(t_1, \dots, D[\ell_i], \dots, t_m)]$ then $g \in \mathcal{A}$

\implies easily adapted to a system with only semi-linear rules

\implies easily adapted to a system on which call-by-value rewriting can be imposed

Overview

- 1 Motivation
- 2 First-order Term Rewriting
- 3 Characterising PTIME
- 4 Call-by-value Rewriting
- 5 Other Restrictions
- 6 Conclusions**

Results:

Results:

- we transposed the characterisation of PTIME in [Jones '01] to **call-by-value** term rewriting

Results:

- we transposed the characterisation of PTIME in [Jones '01] to **call-by-value** term rewriting (including non-determinism)

Results:

- we transposed the characterisation of PTIME in [Jones '01] to **call-by-value** term rewriting (including non-determinism)
- we showed that this characterisation can be used as a **tool**:

Results:

- we transposed the characterisation of PTIME in [Jones '01] to **call-by-value** term rewriting (including non-determinism)
- we showed that this characterisation can be used as a **tool**:
 - simplified proof of [Carvalho and Simonsen '14] through a transformation to call-by-value term rewriting

Results:

- we transposed the characterisation of PTIME in [Jones '01] to **call-by-value** term rewriting (including non-determinism)
- we showed that this characterisation can be used as a **tool**:
 - simplified proof of [Carvalho and Simonsen '14] through a transformation to call-by-value term rewriting
 - suitability of other constraints on term rewriting might be proved similarly

Results:

- we transposed the characterisation of PTIME in [Jones '01] to **call-by-value** term rewriting (including non-determinism)
- we showed that this characterisation can be used as a **tool**:
 - simplified proof of [Carvalho and Simonsen '14] through a transformation to call-by-value term rewriting
 - suitability of other constraints on term rewriting might be proved similarly

Future work:

Results:

- we transposed the characterisation of PTIME in [Jones '01] to **call-by-value** term rewriting (**including non-determinism**)
- we showed that this characterisation can be used as a **tool**:
 - simplified proof of [Carvalho and Simonsen '14] through a transformation to call-by-value term rewriting
 - suitability of other constraints on term rewriting might be proved similarly

Future work:

- extension to higher-order rewriting (transposing **all** of [Jones '01])

Results:

- we transposed the characterisation of PTIME in [Jones '01] to **call-by-value** term rewriting (**including non-determinism**)
- we showed that this characterisation can be used as a **tool**:
 - simplified proof of [Carvalho and Simonsen '14] through a transformation to call-by-value term rewriting
 - suitability of other constraints on term rewriting might be proved similarly

Future work:

- extension to higher-order rewriting (transposing **all** of [Jones '01])
- considering other constraints and no constraints

Results:

- we transposed the characterisation of PTIME in [Jones '01] to **call-by-value** term rewriting (**including non-determinism**)
- we showed that this characterisation can be used as a **tool**:
 - simplified proof of [Carvalho and Simonsen '14] through a transformation to call-by-value term rewriting
 - suitability of other constraints on term rewriting might be proved similarly

Future work:

- extension to higher-order rewriting (transposing **all** of [Jones '01])
- considering other constraints and **no constraints** (**submitted to FSCD'16**)

Results:

- we transposed the characterisation of PTIME in [Jones '01] to **call-by-value** term rewriting (**including non-determinism**)
- we showed that this characterisation can be used as a **tool**:
 - simplified proof of [Carvalho and Simonsen '14] through a transformation to call-by-value term rewriting
 - suitability of other constraints on term rewriting might be proved similarly

Future work:

- extension to higher-order rewriting (transposing **all** of [Jones '01])
- considering other constraints and no constraints (**submitted to FSCD'16**)
- finding constraints which can **not** be transposed to call-by-value rewriting

Results:

- we transposed the characterisation of PTIME in [Jones '01] to **call-by-value** term rewriting (**including non-determinism**)
- we showed that this characterisation can be used as a **tool**:
 - simplified proof of [Carvalho and Simonsen '14] through a transformation to call-by-value term rewriting
 - suitability of other constraints on term rewriting might be proved similarly

Future work:

- extension to higher-order rewriting (transposing **all** of [Jones '01])
- considering other constraints and no constraints (**submitted to FSCD'16**)
- finding constraints which can **not** be transposed to call-by-value rewriting

Questions?