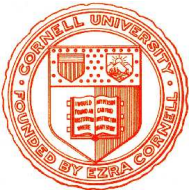


Automatisierte Logik und Programmierung



Lektion 19

Korrektheitserhaltende Optimierungen



1. Logische Vereinfachungen
2. Partielle Auswertung
3. Endliche Differenzierung
4. Fallanalyse
5. Datentyp-Verfeinerung

Eliminiere überflüssige Berechnungen

- **Simplifikation: Logische Vereinfachung**
 - Äquivalenzumwandlung von Teilausdrücken, ggf. im Kontext

Eliminiere überflüssige Berechnungen

- **Simplifikation: Logische Vereinfachung**
 - Äquivalenzumwandlung von Teilausdrücken, ggf. im Kontext
- **Partielle Auswertung**
 - Symbolische Auswertung von Ausdrücken mit konstanten Komponenten

Eliminiere überflüssige Berechnungen

- **Simplifikation: Logische Vereinfachung**
 - Äquivalenzumwandlung von Teilausdrücken, ggf. im Kontext
- **Partielle Auswertung**
 - Symbolische Auswertung von Ausdrücken mit konstanten Komponenten
- **Endliche Differenzierung**
 - Inkrementelle Berechnung von Teilausdrücken in Schleifen

Eliminiere überflüssige Berechnungen

- **Simplifikation: Logische Vereinfachung**
 - Äquivalenzumwandlung von Teilausdrücken, ggf. im Kontext
- **Partielle Auswertung**
 - Symbolische Auswertung von Ausdrücken mit konstanten Komponenten
- **Endliche Differenzierung**
 - Inkrementelle Berechnung von Teilausdrücken in Schleifen
- **Fallanalyse**
 - Analyse und Vereinfachung von Teilausdrücken

Eliminiere überflüssige Berechnungen

- **Simplifikation: Logische Vereinfachung**
 - Äquivalenzumwandlung von Teilausdrücken, ggf. im Kontext
- **Partielle Auswertung**
 - Symbolische Auswertung von Ausdrücken mit konstanten Komponenten
- **Endliche Differenzierung**
 - Inkrementelle Berechnung von Teilausdrücken in Schleifen
- **Fallanalyse**
 - Analyse und Vereinfachung von Teilausdrücken
- **Datentyp-Verfeinerung**
 - Bestimmung konkreter Implementierungen für abstrakte Datentypen

Eliminiere überflüssige Berechnungen

- **Simplifikation: Logische Vereinfachung**
 - Äquivalenzumwandlung von Teilausdrücken, ggf. im Kontext
- **Partielle Auswertung**
 - Symbolische Auswertung von Ausdrücken mit konstanten Komponenten
- **Endliche Differenzierung**
 - Inkrementelle Berechnung von Teilausdrücken in Schleifen
- **Fallanalyse**
 - Analyse und Vereinfachung von Teilausdrücken
- **Datentyp-Verfeinerung**
 - Bestimmung konkreter Implementierungen für abstrakte Datentypen
- **Sprachabhängige Optimierung & Compilierung**
 - Ausnutzen der Besonderheiten einer konkreten Zielsprache

Logische Vereinfachung von Teilausdrücken

- **Transformation in äquivalente Ausdrücke**
 - Term-Rewriting mit gerichteten Gleichungen
 - Gleichungen formuliert als Lemmata der Wissensbank
 - Identifikation geeigneter Lemmata über Operatoren im Ausdruck

Logische Vereinfachung von Teilausdrücken

- **Transformation in äquivalente Ausdrücke**
 - Term-Rewriting mit gerichteten Gleichungen
 - Gleichungen formuliert als Lemmata der Wissensbank
 - Identifikation geeigneter Lemmata über Operatoren im Ausdruck
$$a \in (x. []) \circ (b.L)$$

Logische Vereinfachung von Teilausdrücken

● Transformation in äquivalente Ausdrücke

- Term-Rewriting mit gerichteten Gleichungen
- Gleichungen formuliert als Lemmata der Wissensbank
- Identifikation geeigneter Lemmata über Operatoren im Ausdruck

$$a \in (x. []) \circ (b.L)$$

$$(x.L_1) \circ L_2 = x.(L_1 \circ L_2)$$

append / cons

Logische Vereinfachung von Teilausdrücken

● Transformation in äquivalente Ausdrücke

- Term-Rewriting mit gerichteten Gleichungen
- Gleichungen formuliert als Lemmata der Wissensbank
- Identifikation geeigneter Lemmata über Operatoren im Ausdruck

$$a \in (x. []) \circ (b.L) \mapsto a \in x. ([] \circ (b.L))$$

$$(x.L_1) \circ L_2 = x.(L_1 \circ L_2)$$

append / cons

Logische Vereinfachung von Teilausdrücken

● Transformation in äquivalente Ausdrücke

- Term-Rewriting mit gerichteten Gleichungen
- Gleichungen formuliert als Lemmata der Wissensbank
- Identifikation geeigneter Lemmata über Operatoren im Ausdruck

$$a \in (x. [])^{\circ}(b.L) \mapsto a \in x. ([]^{\circ}(b.L))$$

$$[]^{\circ}L = L$$

append / nil

Logische Vereinfachung von Teilausdrücken

● Transformation in äquivalente Ausdrücke

- Term-Rewriting mit gerichteten Gleichungen
- Gleichungen formuliert als Lemmata der Wissensbank
- Identifikation geeigneter Lemmata über Operatoren im Ausdruck

$$a \in (x. [])^{\circ}(b.L) \mapsto a \in x. ([]^{\circ}(b.L)) \mapsto a \in x. (b.L)$$

$$[]^{\circ}L = L$$

append / nil

Logische Vereinfachung von Teilausdrücken

● Transformation in äquivalente Ausdrücke

- Term-Rewriting mit gerichteten Gleichungen
- Gleichungen formuliert als Lemmata der Wissensbank
- Identifikation geeigneter Lemmata über Operatoren im Ausdruck

$$a \in (x. [])^{\circ} (b.L) \mapsto a \in x.(b.L)$$

$$x \in a.L \Leftrightarrow x=a \vee x \in L$$

member / cons

Logische Vereinfachung von Teilausdrücken

● Transformation in äquivalente Ausdrücke

- Term-Rewriting mit gerichteten Gleichungen
- Gleichungen formuliert als Lemmata der Wissensbank
- Identifikation geeigneter Lemmata über Operatoren im Ausdruck

$$a \in (x. [])^{\circ}(b.L) \mapsto a \in x.(b.L) \mapsto a=x \vee a \in b.L$$

$$x \in a.L \Leftrightarrow x=a \vee x \in L$$

member / cons

Logische Vereinfachung von Teilausdrücken

● Transformation in äquivalente Ausdrücke

- Term-Rewriting mit gerichteten Gleichungen
- Gleichungen formuliert als Lemmata der Wissensbank
- Identifikation geeigneter Lemmata über Operatoren im Ausdruck

$$a \in (x. [])^{\circ}(b.L) \mapsto a=x \vee a \in b.L$$

$$x \in a.L \Leftrightarrow x=a \vee x \in L$$

member / cons

Logische Vereinfachung von Teilausdrücken

● Transformation in äquivalente Ausdrücke

- Term-Rewriting mit gerichteten Gleichungen
- Gleichungen formuliert als Lemmata der Wissensbank
- Identifikation geeigneter Lemmata über Operatoren im Ausdruck

$$a \in (x. [])^{\circ}(b.L) \mapsto a=x \vee a \in b.L \mapsto a=x \vee a=b \vee a \in L$$

$$x \in a.L \Leftrightarrow x=a \vee x \in L$$

member / cons

Logische Vereinfachung von Teilausdrücken

- **Transformation in äquivalente Ausdrücke**

- Term-Rewriting mit gerichteten Gleichungen
- Gleichungen formuliert als Lemmata der Wissensbank
- Identifikation geeigneter Lemmata über Operatoren im Ausdruck

$$a \in (x. [])^{\circ}(b.L) \mapsto a=x \vee a=b \vee a \in L$$

- **Kontextunabhängige CI-Simplifikation**

- Anwendung einfacher Gleichungen ohne Rahmenbedingungen
- Nur der aktuelle Teilausdruck muß betrachtet werden
- Automatische Ausführung solange anwendbare Lemmata vorhanden

Logische Vereinfachung von Teilausdrücken

- **Transformation in äquivalente Ausdrücke**

- Term-Rewriting mit gerichteten Gleichungen
- Gleichungen formuliert als Lemmata der Wissensbank
- Identifikation geeigneter Lemmata über Operatoren im Ausdruck

$$a \in (x. []) \circ (b.L) \mapsto a=x \vee a=b \vee a \in L$$

- **Kontextunabhängige CI-Simplifikation**

- Anwendung einfacher Gleichungen ohne Rahmenbedingungen
- Nur der aktuelle Teilausdruck muß betrachtet werden
- Automatische Ausführung solange anwendbare Lemmata vorhanden

- **Kontextabhängige CD-Simplifikation**

- Anwendung von Gleichungen mit Rahmenbedingungen
- Rahmenbedingungen müssen durch Kontext erfüllt werden

Logische Vereinfachung von Teilausdrücken

- **Transformation in äquivalente Ausdrücke**

- Term-Rewriting mit gerichteten Gleichungen
- Gleichungen formuliert als Lemmata der Wissensbank
- Identifikation geeigneter Lemmata über Operatoren im Ausdruck

$$a \in (x. [])^{\circ}(b.L) \mapsto a=x \vee a=b \vee a \in L$$

- **Kontextunabhängige CI-Simplifikation**

- Anwendung einfacher Gleichungen ohne Rahmenbedingungen
- Nur der aktuelle Teilausdruck muß betrachtet werden
- Automatische Ausführung solange anwendbare Lemmata vorhanden

- **Kontextabhängige CD-Simplifikation**

- Anwendung von Gleichungen mit Rahmenbedingungen
- Rahmenbedingungen müssen durch Kontext erfüllt werden

- **Benutzerinteraktion**

- Auswahl von Teilausdruck und Art der Vereinfachung (CI/CD)
- Optional bei CD: Begrenzung der Vor- und Rückwärtsinferenzen

OPTIMIERUNG DES COSTAS-ARRAYS ALGORITHMUS

Ausgangspunkt: schematischer Globalsuchalgorithmus

```
FUNCTION Costas (n:ℤ) WHERE n ≥ 1
  RETURNS {p:Seq(ℤ) | perm(p, {1..n}) ∧ ∀j < |p|. nodups(dtrow(p, j))}
≡ if nodups([]) ∧ ∀j < |[]|. nodups(dtrow([], j))
   then Costasgs(n, []) else ∅

FUNCTION Costasaux (n, s:ℤ × Seq(ℤ))
  WHERE n ≥ 1 ∧ range(s) ⊆ {1..n} ∧ nodups(s) ∧ ∀j < |s|. nodups(dtrow(s, j))
  RETURNS {p:Seq(ℤ) | perm(p, {1..n}) ∧ s ⊆ p ∧ ∀j < |p|. nodups(dtrow(p, j))}
≡ {p | p ∈ {s} ∧ perm(p, {1..n}) ∧ ∀j < |p|. nodups(dtrow(p, j)) }
  ∪ ⋃ {Costasgs(n, t) | t ∈ {s.i | i ∈ {1..n}} ∧ nodups(t)
                                             ∧ ∀j < |t|. nodups(dtrow(t, j)) }
```

CI-SIMPLIFIKATIONEN IM HAUPTALGORITHMUS **COSTAS**

```
FUNCTION Costas (n:ℤ) WHERE n ≥ 1  
  RETURNS {p:Seq(ℤ) | perm(p, {1..n}) ∧ ∀j < |p|.nodups(dtrow(p, j))}  
≡ if nodups([]) ∧ ∀j < |[]|.nodups(dtrow([], j))  
   then Costasgs(n, []) else ∅
```

CI-SIMPLIFIKATIONEN IM HAUPTALGORITHMUS **COSTAS**

```
FUNCTION Costas (n:ℤ) WHERE n ≥ 1  
  RETURNS {p:Seq(ℤ) | perm(p, {1..n}) ∧ ∀j < |p|. nodups(dtrow(p, j))}  
≡ if nodups([]) ∧ ∀j < |[]|. nodups(dtrow([], j))  
   then Costasgs(n, []) else ∅
```

Domänenwissen: **nodups**([]) ≡ true

CI-SIMPLIFIKATIONEN IM HAUPTALGORITHMUS **COSTAS**

```
FUNCTION Costas (n:ℤ) WHERE n ≥ 1  
  RETURNS {p:Seq(ℤ) | perm(p, {1..n}) ∧ ∀j < |p|.nodups(dtrow(p, j))}  
≡ if nodups([]) ∧ ∀j < |[]|.nodups(dtrow([], j))  
   then Costasgs(n, []) else ∅
```

```
FUNCTION Costas (n:ℤ) WHERE n ≥ 1  
  RETURNS {p:Seq(ℤ) | perm(p, {1..n}) ∧ ∀j < |p|.nodups(dtrow(p, j))}  
≡ if true ∧ ∀j < |[]|.nodups(dtrow([], j))  
   then Costasgs(n, []) else ∅
```

Domänenwissen: `nodups([])` $\equiv$ `true`

CI-SIMPLIFIKATIONEN IM HAUPTALGORITHMUS **COSTAS**

```
FUNCTION Costas (n:ℤ) WHERE n ≥ 1
  RETURNS {p:Seq(ℤ) | perm(p, {1..n}) ∧ ∀j < |p|.nodups(dtrow(p, j))}
≡ if nodups([]) ∧ ∀j < |[]|.nodups(dtrow([], j))
   then Costasgs(n, []) else ∅
```

```
FUNCTION Costas (n:ℤ) WHERE n ≥ 1
  RETURNS {p:Seq(ℤ) | perm(p, {1..n}) ∧ ∀j < |p|.nodups(dtrow(p, j))}
≡ if true ∧ ∀j < |[]|.nodups(dtrow([], j))
   then Costasgs(n, []) else ∅
```

Domänenwissen: $|[]| \equiv 0$

CI-SIMPLIFIKATIONEN IM HAUPTALGORITHMUS **COSTAS**

```
FUNCTION Costas (n:ℤ) WHERE n ≥ 1
  RETURNS {p:Seq(ℤ) | perm(p, {1..n}) ∧ ∀j < |p|.nodups(dtrow(p, j))}
≡ if nodups([]) ∧ ∀j < |[]|.nodups(dtrow([], j))
   then Costasgs(n, []) else ∅
```

```
FUNCTION Costas (n:ℤ) WHERE n ≥ 1
  RETURNS {p:Seq(ℤ) | perm(p, {1..n}) ∧ ∀j < |p|.nodups(dtrow(p, j))}
≡ if true ∧ ∀j < 0.nodups(dtrow([], j))
   then Costasgs(n, []) else ∅
```

Domänenwissen: $|[]| \equiv 0$

Domänenwissen: $\forall x < 0. P[x] \equiv \text{true}$

CI-SIMPLIFIKATIONEN IM HAUPTALGORITHMUS **COSTAS**

```
FUNCTION Costas (n:ℤ) WHERE n ≥ 1
  RETURNS {p:Seq(ℤ) | perm(p, {1..n}) ∧ ∀j < |p|. nodups(dtrow(p, j))}
≡ if nodups([]) ∧ ∀j < |[]|. nodups(dtrow([], j))
   then Costasgs(n, []) else ∅
```

```
FUNCTION Costas (n:ℤ) WHERE n ≥ 1
  RETURNS {p:Seq(ℤ) | perm(p, {1..n}) ∧ ∀j < |p|. nodups(dtrow(p, j))}
≡ if true ∧ true
   then Costasgs(n, []) else ∅
```

Domänenwissen: $|[]| \equiv 0$

Domänenwissen: $\forall x < 0. P[x] \equiv \text{true}$

CI-SIMPLIFIKATIONEN IM HAUPTALGORITHMUS **COSTAS**

```
FUNCTION Costas (n:ℤ) WHERE n ≥ 1  
  RETURNS {p:Seq(ℤ) | perm(p, {1..n}) ∧ ∀j < |p|.nodups(dtrow(p, j))}  
≡ if nodups([]) ∧ ∀j < |[]|.nodups(dtrow([], j))  
   then Costasgs(n, []) else ∅
```

```
FUNCTION Costas (n:ℤ) WHERE n ≥ 1  
  RETURNS {p:Seq(ℤ) | perm(p, {1..n}) ∧ ∀j < |p|.nodups(dtrow(p, j))}  
≡ if true ∧ true  
   then Costasgs(n, []) else ∅
```

Domänenwissen: **true** ∧ **true** ≡ **true**

CI-SIMPLIFIKATIONEN IM HAUPTALGORITHMUS **COSTAS**

```
FUNCTION Costas (n:ℤ) WHERE n ≥ 1  
  RETURNS {p:Seq(ℤ) | perm(p, {1..n}) ∧ ∀j < |p|.nodups(dtrow(p, j))}  
≡ if nodups([]) ∧ ∀j < |[]|.nodups(dtrow([], j))  
   then Costasgs(n, []) else ∅
```

```
FUNCTION Costas (n:ℤ) WHERE n ≥ 1  
  RETURNS {p:Seq(ℤ) | perm(p, {1..n}) ∧ ∀j < |p|.nodups(dtrow(p, j))}  
≡ if true  
   then Costasgs(n, []) else ∅
```

Domänenwissen: **true** ∧ **true** ≡ **true**

CI-SIMPLIFIKATIONEN IM HAUPTALGORITHMUS **COSTAS**

```
FUNCTION Costas (n:ℤ) WHERE n ≥ 1
  RETURNS {p:Seq(ℤ) | perm(p, {1..n}) ∧ ∀j < |p|.nodups(dtrow(p, j))}
≡ if nodups([]) ∧ ∀j < |[]|.nodups(dtrow([], j))
  then Costasgs(n, []) else ∅
```

```
FUNCTION Costas (n:ℤ) WHERE n ≥ 1
  RETURNS {p:Seq(ℤ) | perm(p, {1..n}) ∧ ∀j < |p|.nodups(dtrow(p, j))}
≡ if true
  then Costasgs(n, []) else ∅
```

Domänenwissen: $\text{if true then } a \text{ else } b \equiv a$

CI-SIMPLIFIKATIONEN IM HAUPTALGORITHMUS **COSTAS**

```
FUNCTION Costas (n:ℤ) WHERE n ≥ 1
  RETURNS {p:Seq(ℤ) | perm(p, {1..n}) ∧ ∀j < |p|.nodups(dtrow(p, j))}
≡ if nodups([]) ∧ ∀j < |[]|.nodups(dtrow([], j))
   then Costasgs(n, []) else ∅
```

```
FUNCTION Costas (n:ℤ) WHERE n ≥ 1
  RETURNS {p:Seq(ℤ) | perm(p, {1..n}) ∧ ∀j < |p|.nodups(dtrow(p, j))}
≡ Costasgs(n, [])
```

Domänenwissen: $\text{if true then } a \text{ else } b \equiv a$

CI-SIMPLIFIKATIONEN IM HAUPTALGORITHMUS **COSTAS**

```
FUNCTION Costas (n:ℤ) WHERE n ≥ 1
  RETURNS {p:Seq(ℤ) | perm(p, {1..n}) ∧ ∀j < |p|.nodups(dtrow(p, j))}
≡ if nodups([]) ∧ ∀j < |[]|.nodups(dtrow([], j))
   then Costasgs(n, []) else ∅
```

```
FUNCTION Costas (n:ℤ) WHERE n ≥ 1
  RETURNS {p:Seq(ℤ) | perm(p, {1..n}) ∧ ∀j < |p|.nodups(dtrow(p, j))}
≡ Costasgs(n, [])
```

CI-SIMPLIFIKATIONEN IN DER HILFSFUNKTION Costas_{aux}

FUNCTION Costas_{aux} ($n, s: \mathbb{Z} \times \text{Seq}(\mathbb{Z})$)
WHERE $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s, j))$
RETURNS $\{p: \text{Seq}(\mathbb{Z}) \mid \text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p, j))\}$
 $\equiv$ $\{p \mid p \in \{s\} \wedge \text{perm}(p, \{1..n\}) \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p, j))\}$
 $\cup \bigcup \{ \text{Costas}_{gs}(n, t) \mid t \in \{s \cdot i \mid i \in \{1..n\}\} \wedge \text{nodups}(t) \wedge \forall j < |t|. \text{nodups}(\text{dtrow}(t, j)) \}$

CI-SIMPLIFIKATIONEN IN DER HILFSFUNKTION Costas_{aux}

FUNCTION $\text{Costas}_{aux} (n, s: \mathbb{Z} \times \text{Seq}(\mathbb{Z}))$
 WHERE $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s, j))$
 RETURNS $\{p: \text{Seq}(\mathbb{Z}) \mid \text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p, j))\}$
 $\equiv$ $\{p \mid p \in \{s\} \wedge \text{perm}(p, \{1..n\}) \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p, j))\}$
 $\cup \bigcup \{ \text{Costas}_{gs}(n, t) \mid t \in \{s \cdot i \mid i \in \{1..n\}\} \wedge \text{nodups}(t)$
 $\wedge \forall j < |t|. \text{nodups}(\text{dtrow}(t, j)) \}$

Domänenwissen: $\{z \mid z \in \{x\} \wedge P[z]\} \equiv \text{if } P[x] \text{ then } \{x\} \text{ else } \emptyset$

CI-SIMPLIFIKATIONEN IN DER HILFSFUNKTION Costas_{aux}

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$ 
   $\{p \mid p \in \{s\} \wedge \text{perm}(p,\{1..n\}) \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
   $\cup \bigcup \{ \text{Costas}_{gs}(n,t) \mid t \in \{s \cdot i \mid i \in \{1..n\}\} \wedge \text{nodups}(t)$ 
   $\wedge \forall j < |t|. \text{nodups}(\text{dtrow}(t,j)) \}$ 

```

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$ 
  if  $\text{perm}(s,\{1..n\}) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$  then  $\{s\}$  else  $\emptyset$ 
   $\cup \bigcup \{ \text{Costas}_{gs}(n,t) \mid t \in \{s \cdot i \mid i \in \{1..n\}\} \wedge \text{nodups}(t)$ 
   $\wedge \forall j < |t|. \text{nodups}(\text{dtrow}(t,j)) \}$ 

```

Domänenwissen: $\{z \mid z \in \{x\} \wedge P[z]\} \equiv \text{if } P[x] \text{ then } \{x\} \text{ else } \emptyset$

CI-SIMPLIFIKATIONEN IN DER HILFSFUNKTION Costas_{aux}

FUNCTION $\text{Costas}_{aux} (n, s: \mathbb{Z} \times \text{Seq}(\mathbb{Z}))$
 WHERE $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s, j))$
 RETURNS $\{p: \text{Seq}(\mathbb{Z}) \mid \text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p, j))\}$
 $\equiv$ $\{p \mid p \in \{s\} \wedge \text{perm}(p, \{1..n\}) \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p, j))\}$
 $\cup \bigcup \{ \text{Costas}_{gs}(n, t) \mid t \in \{s \cdot i \mid i \in \{1..n\}\} \wedge \text{nodups}(t)$
 $\wedge \forall j < |t|. \text{nodups}(\text{dtrow}(t, j)) \}$

FUNCTION $\text{Costas}_{aux} (n, s: \mathbb{Z} \times \text{Seq}(\mathbb{Z}))$
 WHERE $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s, j))$
 RETURNS $\{p: \text{Seq}(\mathbb{Z}) \mid \text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p, j))\}$
 $\equiv$ if $\text{perm}(s, \{1..n\}) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s, j))$ then $\{s\}$ else $\emptyset$
 $\cup \bigcup \{ \text{Costas}_{gs}(n, t) \mid t \in \{s \cdot i \mid i \in \{1..n\}\} \wedge \text{nodups}(t)$
 $\wedge \forall j < |t|. \text{nodups}(\text{dtrow}(t, j)) \}$

Domänenwissen: $\{ f[x, t] \mid t \in \{g[x, y] \mid y \in S\} \wedge h[t] \}$
 $\equiv \{ f[x, g[x, y]] \mid y \in S \wedge h[g[x, y]] \}$

CI-SIMPLIFIKATIONEN IN DER HILFSFUNKTION Costas_{aux}

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$ 
   $\{p \mid p \in \{s\} \wedge \text{perm}(p,\{1..n\}) \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\cup \bigcup \{ \text{Costas}_{gs}(n,t) \mid t \in \{s \cdot i \mid i \in \{1..n\}\} \wedge \text{nodups}(t)$ 
 $\wedge \forall j < |t|. \text{nodups}(\text{dtrow}(t,j)) \}$ 

```

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$ 
  if  $\text{perm}(s,\{1..n\}) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$  then  $\{s\}$  else  $\emptyset$ 
 $\cup \bigcup \{ \text{Costas}_{gs}(n,s \cdot i) \mid i \in \{1..n\} \wedge \text{nodups}(s \cdot i)$ 
 $\wedge \forall j < |s \cdot i|. \text{nodups}(\text{dtrow}(s \cdot i,j)) \}$ 

```

Domänenwissen: $\{ f[x,t] \mid t \in \{ g[x,y] \mid y \in S \} \wedge h[t] \}$

$\equiv \{ f[x, g[x,y]] \mid y \in S \wedge h[g[x,y]] \}$

CI-SIMPLIFIKATIONEN IN DER HILFSFUNKTION Costas_{aux}

```
FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )  
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$   
  RETURNS {p:Seq( $\mathbb{Z}$ ) | perm(p,{1..n})  $\wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))$ }  
 $\equiv$  {p | p  $\in \{s\} \wedge \text{perm}(p,\{1..n\}) \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))$ }  
   $\cup \bigcup \{ \text{Costas}_{gs}(n,t) \mid t \in \{s \cdot i \mid i \in \{1..n\}\} \wedge \text{nodups}(t)$   
                                            $\wedge \forall j < |t|. \text{nodups}(\text{dtrow}(t,j)) \}$ 
```

```
FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )  
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$   
  RETURNS {p:Seq( $\mathbb{Z}$ ) | perm(p,{1..n})  $\wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))$ }  
 $\equiv$  if perm(s,{1..n})  $\wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$  then {s} else  $\emptyset$   
   $\cup \bigcup \{ \text{Costas}_{gs}(n,s \cdot i) \mid i \in \{1..n\} \wedge \text{nodups}(s \cdot i)$   
                                            $\wedge \forall j < |s \cdot i|. \text{nodups}(\text{dtrow}(s \cdot i,j)) \}$ 
```

Vereinfachung unter Berücksichtigung von Kontext

- **Anwendung bedingter Gleichungen**

- Gleichung $A \equiv B$ mit Vorbedingung C
- Formuliert als Lemma der Form $C \Rightarrow A \equiv B$

Vereinfachung unter Berücksichtigung von Kontext

- **Anwendung bedingter Gleichungen**

- Gleichung $A \equiv B$ mit Vorbedingung C
- Formuliert als Lemma der Form $C \Rightarrow A \equiv B$

- **Anwendbarkeit abhängig vom Kontext**

- Ersetze Teilausdruck A durch B , wenn Kontext Bedingung C erfüllt
- Kontext ergibt sich aus Syntaxbaum des Gesamtausdrucks
 - Bei Programmen: benachbarter Programmcode und Vorbedingung
- Hauptanwendung: Elimination redundanter Teilausdrücke

Vereinfachung unter Berücksichtigung von Kontext

- **Anwendung bedingter Gleichungen**

- Gleichung $A \equiv B$ mit Vorbedingung C
- Formuliert als Lemma der Form $C \Rightarrow A \equiv B$

- **Anwendbarkeit abhängig vom Kontext**

- Ersetze Teilausdruck A durch B , wenn Kontext Bedingung C erfüllt
- Kontext ergibt sich aus Syntaxbaum des Gesamtausdrucks
 - Bei Programmen: benachbarter Programmcode und Vorbedingung
- Hauptanwendung: Elimination redundanter Teilausdrücke

- **Komplizierter als CI-Simplifikation**

- Vorbedingung kann auch Teil einer Gleichung $C \wedge A \equiv B$ sein
- Gleichung kann oft auch von rückwärts angewandt werden
- Automatische Anwendung muß tiefenbeschränkt werden

CD-SIMPLIFIKATIONEN IN DER HILFSFUNKTION Costas_{aux}

```
FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )  
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$   
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$   
 $\equiv$       if  $\text{perm}(s,\{1..n\}) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$  then  $\{s\}$  else  $\emptyset$   
       $\cup \bigcup \{ \text{Costas}_{gs}(n,s \cdot i) \mid i \in \{1..n\} \wedge \text{nodups}(s \cdot i)$   
           $\wedge \forall j < |s \cdot i|. \text{nodups}(\text{dtrow}(s \cdot i,j)) \}$ 
```

CD-SIMPLIFIKATIONEN IN DER HILFSFUNKTION Costas_{aux}

```
FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )  
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$   
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$   
 $\equiv$       if  $\text{perm}(s,\{1..n\}) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$  then  $\{s\}$  else  $\emptyset$   
       $\cup \bigcup \{ \text{Costas}_{gs}(n,s \cdot i) \mid i \in \{1..n\} \wedge \text{nodups}(s \cdot i)$   
           $\wedge \forall j < |s \cdot i|. \text{nodups}(\text{dtrow}(s \cdot i,j)) \}$ 
```

Domänenwissen: $P \Rightarrow (Q \wedge P \equiv Q)$

Kontext der Vorbedingung: $\forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$

CD-SIMPLIFIKATIONEN IN DER HILFSFUNKTION Costas_{aux}

```
FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )  
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$   
  RETURNS {p: $\text{Seq}(\mathbb{Z}) \mid \text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))$ }  
 $\equiv$     if  $\text{perm}(s, \{1..n\}) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$  then {s} else  $\emptyset$   
     $\cup \bigcup \{ \text{Costas}_{gs}(n, s \cdot i) \mid i \in \{1..n\} \wedge \text{nodups}(s \cdot i)$   
         $\wedge \forall j < |s \cdot i|. \text{nodups}(\text{dtrow}(s \cdot i, j)) \}$ 
```

```
FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )  
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$   
  RETURNS {p: $\text{Seq}(\mathbb{Z}) \mid \text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))$ }  
 $\equiv$     if  $\text{perm}(s, \{1..n\})$  then {s} else  $\emptyset$   
     $\cup \bigcup \{ \text{Costas}_{gs}(n, s \cdot i) \mid i \in \{1..n\} \wedge \text{nodups}(s \cdot i)$   
         $\wedge \forall j < |s \cdot i|. \text{nodups}(\text{dtrow}(s \cdot i, j)) \}$ 
```

Domänenwissen: $P \Rightarrow (Q \wedge P \equiv Q)$

Kontext der Vorbedingung: $\forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$

CD-SIMPLIFIKATIONEN IN DER HILFSFUNKTION Costas_{aux}

```
FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )  
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$   
  RETURNS {p: $\text{Seq}(\mathbb{Z})$  |  $\text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))$ }  
 $\equiv$  if  $\text{perm}(s, \{1..n\}) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$  then {s} else  $\emptyset$   
   $\cup \bigcup \{ \text{Costas}_{gs}(n, s.i) \mid i \in \{1..n\} \wedge \text{nodups}(s.i) \wedge \forall j < |s.i|. \text{nodups}(\text{dtrow}(s.i,j)) \}$ 
```

```
FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )  
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$   
  RETURNS {p: $\text{Seq}(\mathbb{Z})$  |  $\text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))$ }  
 $\equiv$  if  $\text{perm}(s, \{1..n\})$  then {s} else  $\emptyset$   
   $\cup \bigcup \{ \text{Costas}_{gs}(n, s.i) \mid i \in \{1..n\} \wedge \text{nodups}(s.i) \wedge \forall j < |s.i|. \text{nodups}(\text{dtrow}(s.i,j)) \}$ 
```

Domänenwissen: $\text{perm}(L, M) \equiv \text{range}(L) \subseteq M \wedge M \subseteq \text{range}(L) \wedge \text{nodups}(L)$

Kontext der Vorbedingung: $\text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s)$

CD-SIMPLIFIKATIONEN IN DER HILFSFUNKTION Costas_{aux}

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$     if  $\text{perm}(s,\{1..n\}) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$  then  $\{s\}$  else  $\emptyset$ 
       $\cup \bigcup \{ \text{Costas}_{gs}(n,s \cdot i) \mid i \in \{1..n\} \wedge \text{nodups}(s \cdot i)$ 
                                      $\wedge \forall j < |s \cdot i|. \text{nodups}(\text{dtrow}(s \cdot i,j)) \}$ 

```

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$     if  $\{1..n\} \subseteq \text{range}(s)$  then  $\{s\}$  else  $\emptyset$ 
       $\cup \bigcup \{ \text{Costas}_{gs}(n,s \cdot i) \mid i \in \{1..n\} \wedge \text{nodups}(s \cdot i)$ 
                                      $\wedge \forall j < |s \cdot i|. \text{nodups}(\text{dtrow}(s \cdot i,j)) \}$ 

```

Domänenwissen: $\text{perm}(L,M) \equiv \text{range}(L) \subseteq M \wedge M \subseteq \text{range}(L) \wedge \text{nodups}(L)$

Kontext der Vorbedingung: $\text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s)$

Domänenwissen: $M \subseteq M' \equiv M \setminus M' = \emptyset$

CD-SIMPLIFIKATIONEN IN DER HILFSFUNKTION Costas_{aux}

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS {p: $\text{Seq}(\mathbb{Z})$  |  $\text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))$ }
 $\equiv$     if  $\text{perm}(s, \{1..n\}) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$  then {s} else  $\emptyset$ 
       $\cup \bigcup \{ \text{Costas}_{gs}(n, s \cdot i) \mid i \in \{1..n\} \wedge \text{nodups}(s \cdot i) \wedge \forall j < |s \cdot i|. \text{nodups}(\text{dtrow}(s \cdot i, j)) \}$ 

```

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS {p: $\text{Seq}(\mathbb{Z})$  |  $\text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))$ }
 $\equiv$     if  $\{1..n\} \setminus \text{range}(s) = \emptyset$  then {s} else  $\emptyset$ 
       $\cup \bigcup \{ \text{Costas}_{gs}(n, s \cdot i) \mid i \in \{1..n\} \wedge \text{nodups}(s \cdot i) \wedge \forall j < |s \cdot i|. \text{nodups}(\text{dtrow}(s \cdot i, j)) \}$ 

```

Domänenwissen: $\text{perm}(L, M) \equiv \text{range}(L) \subseteq M \wedge M \subseteq \text{range}(L) \wedge \text{nodups}(L)$

Kontext der Vorbedingung: $\text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s)$

Domänenwissen: $M \subseteq M' \equiv M \setminus M' = \emptyset$

CD-SIMPLIFIKATIONEN IN DER HILFSFUNKTION Costas_{aux}

```
FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )  
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$   
  RETURNS {p: $\text{Seq}(\mathbb{Z})$  |  $\text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))$ }  
 $\equiv$     if  $\text{perm}(s, \{1..n\}) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$  then {s} else  $\emptyset$   
     $\cup \bigcup \{ \text{Costas}_{gs}(n, s \cdot i) \mid i \in \{1..n\} \wedge \text{nodups}(s \cdot i)$   
         $\wedge \forall j < |s \cdot i|. \text{nodups}(\text{dtrow}(s \cdot i, j)) \}$ 
```

```
FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )  
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$   
  RETURNS {p: $\text{Seq}(\mathbb{Z})$  |  $\text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))$ }  
 $\equiv$     if  $\{1..n\} \setminus \text{range}(s) = \emptyset$  then {s} else  $\emptyset$   
     $\cup \bigcup \{ \text{Costas}_{gs}(n, s \cdot i) \mid i \in \{1..n\} \wedge \text{nodups}(s \cdot i)$   
         $\wedge \forall j < |s \cdot i|. \text{nodups}(\text{dtrow}(s \cdot i, j)) \}$ 
```

Domänenwissen: $\text{nodups}(L \cdot x) \equiv \text{nodups}(L) \wedge x \notin L$

Kontext der Vorbedingung: $\text{nodups}(s)$

CD-SIMPLIFIKATIONEN IN DER HILFSFUNKTION Costas_{aux}

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$     if  $\text{perm}(s,\{1..n\}) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$  then  $\{s\}$  else  $\emptyset$ 
       $\cup \bigcup \{ \text{Costas}_{gs}(n,s \cdot i) \mid i \in \{1..n\} \wedge \text{nodups}(s \cdot i)$ 
                                      $\wedge \forall j < |s \cdot i|. \text{nodups}(\text{dtrow}(s \cdot i,j)) \}$ 

```

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$     if  $\{1..n\} \setminus \text{range}(s) = \emptyset$  then  $\{s\}$  else  $\emptyset$ 
       $\cup \bigcup \{ \text{Costas}_{gs}(n,s \cdot i) \mid i \in \{1..n\} \wedge i \notin s$ 
                                      $\wedge \forall j < |s \cdot i|. \text{nodups}(\text{dtrow}(s \cdot i,j)) \}$ 

```

Domänenwissen: $\text{nodups}(L \cdot x) \equiv \text{nodups}(L) \wedge x \notin L$

Kontext der Vorbedingung: $\text{nodups}(s)$

Domänenwissen: $x \in M \wedge x \notin L \equiv x \in M \setminus \text{range}(L)$

CD-SIMPLIFIKATIONEN IN DER HILFSFUNKTION Costas_{aux}

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$     if  $\text{perm}(s,\{1..n\}) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$  then  $\{s\}$  else  $\emptyset$ 
       $\cup \bigcup \{ \text{Costas}_{gs}(n,s \cdot i) \mid i \in \{1..n\} \wedge \text{nodups}(s \cdot i)$ 
                                      $\wedge \forall j < |s \cdot i|. \text{nodups}(\text{dtrow}(s \cdot i,j)) \}$ 

```

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$     if  $\{1..n\} \setminus \text{range}(s) = \emptyset$  then  $\{s\}$  else  $\emptyset$ 
       $\cup \bigcup \{ \text{Costas}_{gs}(n,s \cdot i) \mid i \in \{1..n\} \setminus \text{range}(s)$ 
                                      $\wedge \forall j < |s \cdot i|. \text{nodups}(\text{dtrow}(s \cdot i,j)) \}$ 

```

Domänenwissen: $\text{nodups}(L \cdot x) \equiv \text{nodups}(L) \wedge x \notin L$

Kontext der Vorbedingung: $\text{nodups}(s)$

Domänenwissen: $x \in M \wedge x \notin L \equiv x \in M \setminus \text{range}(L)$

CD-SIMPLIFIKATIONEN IN DER HILFSFUNKTION Costas_{aux}

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS {p: $\text{Seq}(\mathbb{Z})$  |  $\text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))$ }
 $\equiv$     if  $\text{perm}(s, \{1..n\}) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$  then {s} else  $\emptyset$ 
       $\cup \bigcup \{ \text{Costas}_{gs}(n, s \cdot i) \mid i \in \{1..n\} \wedge \text{nodups}(s \cdot i)$ 
                                      $\wedge \forall j < |s \cdot i|. \text{nodups}(\text{dtrow}(s \cdot i, j)) \}$ 

```

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS {p: $\text{Seq}(\mathbb{Z})$  |  $\text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))$ }
 $\equiv$     if  $\{1..n\} \setminus \text{range}(s) = \emptyset$  then {s} else  $\emptyset$ 
       $\cup \bigcup \{ \text{Costas}_{gs}(n, s \cdot i) \mid i \in \{1..n\} \setminus \text{range}(s)$ 
                                      $\wedge \forall j < |s \cdot i|. \text{nodups}(\text{dtrow}(s \cdot i, j)) \}$ 

```

Domänenwissen: $\text{dtrow}(L \cdot x, i) \equiv \text{dtrow}(L, i) \cdot (L_{|L \cdot x| - i} - x)$

CD-SIMPLIFIKATIONEN IN DER HILFSFUNKTION Costas_{aux}

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$     if  $\text{perm}(s,\{1..n\}) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$  then  $\{s\}$  else  $\emptyset$ 
       $\cup \bigcup \{ \text{Costas}_{gs}(n,s \cdot i) \mid i \in \{1..n\} \wedge \text{nodups}(s \cdot i)$ 
                                      $\wedge \forall j < |s \cdot i|. \text{nodups}(\text{dtrow}(s \cdot i,j)) \}$ 

```

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$     if  $\{1..n\} \setminus \text{range}(s) = \emptyset$  then  $\{s\}$  else  $\emptyset$ 
       $\cup \bigcup \{ \text{Costas}_{gs}(n,s \cdot i) \mid i \in \{1..n\} \setminus \text{range}(s)$ 
                                      $\wedge \forall j < |s \cdot i|. \text{nodups}(\text{dtrow}(s,j) \cdot (s_{|s \cdot i| - j} - i)) \}$ 

```

Domänenwissen: $\text{dtrow}(L \cdot x, i) \equiv \text{dtrow}(L, i) \cdot (L_{|L \cdot x| - i} - x)$
 $\text{nodups}(L \cdot x) \equiv \text{nodups}(L) \wedge x \notin L$

CD-SIMPLIFIKATIONEN IN DER HILFSFUNKTION Costas_{aux}

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$ 
  if  $\text{perm}(s,\{1..n\}) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$  then  $\{s\}$  else  $\emptyset$ 
   $\cup \bigcup \{ \text{Costas}_{gs}(n,s \cdot i) \mid i \in \{1..n\} \wedge \text{nodups}(s \cdot i)$ 
     $\wedge \forall j < |s \cdot i|. \text{nodups}(\text{dtrow}(s \cdot i,j)) \}$ 

```

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$ 
  if  $\{1..n\} \setminus \text{range}(s) = \emptyset$  then  $\{s\}$  else  $\emptyset$ 
   $\cup \bigcup \{ \text{Costas}_{gs}(n,s \cdot i) \mid i \in \{1..n\} \setminus \text{range}(s)$ 
     $\wedge \forall j < |s \cdot i|. \text{nodups}(\text{dtrow}(s,j)) \wedge s_{|s \cdot i| - j} - i \notin \text{dtrow}(s,j) \}$ 

```

Domänenwissen: $\text{nodups}(L \cdot x) \equiv \text{nodups}(L) \wedge x \notin L$
 $\forall i < |L \cdot x|. P[i] \equiv \forall i < |L|. P[i] \wedge P[|L|]$

CD-SIMPLIFIKATIONEN IN DER HILFSFUNKTION Costas_{aux}

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$     if  $\text{perm}(s,\{1..n\}) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$  then  $\{s\}$  else  $\emptyset$ 
       $\cup \bigcup \{ \text{Costas}_{gs}(n,s \cdot i) \mid i \in \{1..n\} \wedge \text{nodups}(s \cdot i)$ 
                                      $\wedge \forall j < |s \cdot i|. \text{nodups}(\text{dtrow}(s \cdot i,j)) \}$ 

```

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$     if  $\{1..n\} \setminus \text{range}(s) = \emptyset$  then  $\{s\}$  else  $\emptyset$ 
       $\cup \bigcup \{ \text{Costas}_{gs}(n,s \cdot i) \mid i \in \{1..n\} \setminus \text{range}(s)$ 
                                      $\wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j)) \wedge s_{|s| \cdot i - j - i} \notin \text{dtrow}(s,j)$ 
                                      $\wedge \text{nodups}(\text{dtrow}(s, |s|)) \wedge s_{|s| \cdot i - j - i} \notin \text{dtrow}(s, |s|) \}$ 

```

Domänenwissen: $\forall i < |L \cdot x|. P[i] \equiv \forall i < |L|. P[i] \wedge P[|L|]$
 $\text{dtrow}(L, |L|) \equiv []$

CD-SIMPLIFIKATIONEN IN DER HILFSFUNKTION Costas_{aux}

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$ 
  if  $\text{perm}(s,\{1..n\}) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$  then  $\{s\}$  else  $\emptyset$ 
   $\cup \bigcup \{ \text{Costas}_{gs}(n,s \cdot i) \mid i \in \{1..n\} \wedge \text{nodups}(s \cdot i) \wedge \forall j < |s \cdot i|. \text{nodups}(\text{dtrow}(s \cdot i,j)) \}$ 

```

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$ 
  if  $\{1..n\} \setminus \text{range}(s) = \emptyset$  then  $\{s\}$  else  $\emptyset$ 
   $\cup \bigcup \{ \text{Costas}_{gs}(n,s \cdot i) \mid i \in \{1..n\} \setminus \text{range}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j)) \wedge s_{|s| \cdot i - j} - i \notin \text{dtrow}(s,j) \wedge \text{nodups}([]) \wedge s_{|s| \cdot i - j} - i \notin [] \}$ 

```

Domänenwissen:

$$\text{dtrow}(L, |L|) \equiv []$$

$$\text{nodups}([]) \equiv \text{true}$$

$$x \notin [] \equiv \text{true}$$

CD-SIMPLIFIKATIONEN IN DER HILFSFUNKTION Costas_{aux}

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS {p: $\text{Seq}(\mathbb{Z})$  |  $\text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))$ }
 $\equiv$     if  $\text{perm}(s, \{1..n\}) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$  then {s} else  $\emptyset$ 
       $\cup \bigcup \{ \text{Costas}_{gs}(n, s \cdot i) \mid i \in \{1..n\} \wedge \text{nodups}(s \cdot i)$ 
           $\wedge \forall j < |s \cdot i|. \text{nodups}(\text{dtrow}(s \cdot i, j)) \}$ 

```

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS {p: $\text{Seq}(\mathbb{Z})$  |  $\text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))$ }
 $\equiv$     if  $\{1..n\} \setminus \text{range}(s) = \emptyset$  then {s} else  $\emptyset$ 
       $\cup \bigcup \{ \text{Costas}_{gs}(n, s \cdot i) \mid i \in \{1..n\} \setminus \text{range}(s)$ 
           $\wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j)) \wedge s_{|s \cdot i| - j - i} \notin \text{dtrow}(s,j) \}$ 

```

Domänenwissen:

$$\text{nodups}([]) \equiv \text{true}$$

$$x \notin [] \equiv \text{true}$$

$$\forall x. P[x] \wedge Q[x] \equiv \forall x. P[x] \wedge \forall x. Q[x]$$

CD-SIMPLIFIKATIONEN IN DER HILFSFUNKTION Costas_{aux}

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$ 
  if  $\text{perm}(s,\{1..n\}) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$  then  $\{s\}$  else  $\emptyset$ 
   $\cup \bigcup \{ \text{Costas}_{gs}(n,s \cdot i) \mid i \in \{1..n\} \wedge \text{nodups}(s \cdot i) \wedge \forall j < |s \cdot i|. \text{nodups}(\text{dtrow}(s \cdot i,j)) \}$ 

```

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$ 
  if  $\{1..n\} \setminus \text{range}(s) = \emptyset$  then  $\{s\}$  else  $\emptyset$ 
   $\cup \bigcup \{ \text{Costas}_{gs}(n,s \cdot i) \mid i \in \{1..n\} \setminus \text{range}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j)) \wedge \forall j < |s|. s_{|s|-j-i} \notin \text{dtrow}(s,j) \}$ 

```

Domänenwissen: $\forall x. P[x] \wedge Q[x] \equiv \forall x. P[x] \wedge \forall x. Q[x]$

CD-SIMPLIFIKATIONEN IN DER HILFSFUNKTION Costas_{aux}

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS {p: $\text{Seq}(\mathbb{Z})$  |  $\text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))$ }
 $\equiv$     if  $\text{perm}(s, \{1..n\}) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$  then {s} else  $\emptyset$ 
       $\cup \bigcup \{ \text{Costas}_{gs}(n, s \cdot i) \mid i \in \{1..n\} \wedge \text{nodups}(s \cdot i)$ 
           $\wedge \forall j < |s \cdot i|. \text{nodups}(\text{dtrow}(s \cdot i, j)) \}$ 

```

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS {p: $\text{Seq}(\mathbb{Z})$  |  $\text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))$ }
 $\equiv$     if  $\{1..n\} \setminus \text{range}(s) = \emptyset$  then {s} else  $\emptyset$ 
       $\cup \bigcup \{ \text{Costas}_{gs}(n, s \cdot i) \mid i \in \{1..n\} \setminus \text{range}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
           $\wedge \forall j < |s|. s_{|s| \cdot i - j - i} \notin \text{dtrow}(s,j) \}$ 

```

Kontext der Vorbedingung: $\forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$

CD-SIMPLIFIKATIONEN IN DER HILFSFUNKTION Costas_{aux}

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$     if  $\text{perm}(s,\{1..n\}) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$  then  $\{s\}$  else  $\emptyset$ 
       $\cup \bigcup \{ \text{Costas}_{gs}(n,s \cdot i) \mid i \in \{1..n\} \wedge \text{nodups}(s \cdot i)$ 
                                      $\wedge \forall j < |s \cdot i|. \text{nodups}(\text{dtrow}(s \cdot i,j)) \}$ 

```

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$     if  $\{1..n\} \setminus \text{range}(s) = \emptyset$  then  $\{s\}$  else  $\emptyset$ 
       $\cup \bigcup \{ \text{Costas}_{gs}(n,s \cdot i) \mid i \in \{1..n\} \setminus \text{range}(s) \wedge \forall j < |s|. s_{|s|+j} - i \notin \text{dtrow}(s,j) \}$ 

```

Kontext der Vorbedingung: $\forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$

CD-SIMPLIFIKATIONEN IN DER HILFSFUNKTION Costas_{aux}

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$     if  $\text{perm}(s,\{1..n\}) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$  then  $\{s\}$  else  $\emptyset$ 
       $\cup \bigcup \{ \text{Costas}_{gs}(n,s \cdot i) \mid i \in \{1..n\} \wedge \text{nodups}(s \cdot i)$ 
           $\wedge \forall j < |s \cdot i|. \text{nodups}(\text{dtrow}(s \cdot i,j)) \}$ 

```

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$     if  $\{1..n\} \setminus \text{range}(s) = \emptyset$  then  $\{s\}$  else  $\emptyset$ 
       $\cup \bigcup \{ \text{Costas}_{gs}(n,s \cdot i) \mid i \in \{1..n\} \setminus \text{range}(s) \wedge \forall j < |s|. s_{|s \cdot i| - j} - i \notin \text{dtrow}(s,j) \}$ 

```

Auswertung von Ausdrücken mit Konstanten

- **Symbolische Auswertung zur Entwurfszeit**
 - **Reduktion** von Ausdrücken, deren “Wert” bestimmt werden kann
 - Anwendbar, solange ein konstanter Teilausdruck vorhanden
 - Formale Technik: **Auffalten von Definitionen** + **Simplifikation**
Je nach Inhalt der Wissensbank auch reine Simplifikation

Auswertung von Ausdrücken mit Konstanten

- **Symbolische Auswertung zur Entwurfszeit**
 - **Reduktion** von Ausdrücken, deren “Wert” bestimmt werden kann
 - Anwendbar, solange ein konstanter Teilausdruck vorhanden
 - Formale Technik: **Auffalten von Definitionen** + **Simplifikation**
Je nach Inhalt der Wissensbank auch reine Simplifikation

Beispiel:

$| [x; 5] \circ L |$

unfold append

Auswertung von Ausdrücken mit Konstanten

- **Symbolische Auswertung zur Entwurfszeit**
 - **Reduktion** von Ausdrücken, deren “Wert” bestimmt werden kann
 - Anwendbar, solange ein konstanter Teilausdruck vorhanden
 - Formale Technik: **Auffalten von Definitionen** + **Simplifikation**
Je nach Inhalt der Wissensbank auch reine Simplifikation

Beispiel:

$ [x; 5] \circ L $	unfold append
$\mapsto x . ([5] \circ L) $	unfold append

Auswertung von Ausdrücken mit Konstanten

- **Symbolische Auswertung zur Entwurfszeit**
 - **Reduktion** von Ausdrücken, deren “Wert” bestimmt werden kann
 - Anwendbar, solange ein konstanter Teilausdruck vorhanden
 - Formale Technik: **Auffalten von Definitionen** + **Simplifikation**
Je nach Inhalt der Wissensbank auch reine Simplifikation

Beispiel:

$ [x; 5] \circ L $	unfold append
$\mapsto x . ([5] \circ L) $	unfold append
$\mapsto x . (5 . ([] \circ L)) $	unfold append

Auswertung von Ausdrücken mit Konstanten

- **Symbolische Auswertung zur Entwurfszeit**
 - **Reduktion** von Ausdrücken, deren “Wert” bestimmt werden kann
 - Anwendbar, solange ein konstanter Teilausdruck vorhanden
 - Formale Technik: **Auffalten von Definitionen** + **Simplifikation**
Je nach Inhalt der Wissensbank auch reine Simplifikation

Beispiel:

$ [x; 5] \circ L $	unfold append
$\mapsto x . ([5] \circ L) $	unfold append
$\mapsto x . (5 . ([] \circ L)) $	unfold append
$\mapsto x . (5 . L) $	unfold length

Auswertung von Ausdrücken mit Konstanten

- **Symbolische Auswertung zur Entwurfszeit**
 - Reduktion von Ausdrücken, deren “Wert” bestimmt werden kann
 - Anwendbar, solange ein konstanter Teilausdruck vorhanden
 - Formale Technik: Auffalten von Definitionen + Simplifikation
Je nach Inhalt der Wissensbank auch reine Simplifikation

Beispiel:

$ [x; 5] \circ L $	unfold append
$\mapsto x . ([5] \circ L) $	unfold append
$\mapsto x . (5 . ([] \circ L)) $	unfold append
$\mapsto x . (5 . L) $	unfold length
$\mapsto 1 + 5 . L $	unfold length

Auswertung von Ausdrücken mit Konstanten

- **Symbolische Auswertung zur Entwurfszeit**
 - **Reduktion** von Ausdrücken, deren “Wert” bestimmt werden kann
 - Anwendbar, solange ein konstanter Teilausdruck vorhanden
 - Formale Technik: **Auffalten von Definitionen** + **Simplifikation**
Je nach Inhalt der Wissensbank auch reine Simplifikation

Beispiel:

$ [x; 5] \circ L $	unfold append
$\mapsto x . ([5] \circ L) $	unfold append
$\mapsto x . (5 . ([] \circ L)) $	unfold append
$\mapsto x . (5 . L) $	unfold length
$\mapsto 1 + 5 . L $	unfold length
$\mapsto 1 + 1 + L $	simplify

Auswertung von Ausdrücken mit Konstanten

- **Symbolische Auswertung zur Entwurfszeit**
 - **Reduktion** von Ausdrücken, deren “Wert” bestimmt werden kann
 - Anwendbar, solange ein konstanter Teilausdruck vorhanden
 - Formale Technik: **Auffalten von Definitionen** + **Simplifikation**
Je nach Inhalt der Wissensbank auch reine Simplifikation

Beispiel:

$ [x;5] \circ L $	unfold append
$\mapsto x . ([5] \circ L) $	unfold append
$\mapsto x . (5 . ([] \circ L)) $	unfold append
$\mapsto x . (5 . L) $	unfold length
$\mapsto 1 + 5 . L $	unfold length
$\mapsto 1 + 1 + L $	simplify
$\mapsto 2 + L $	

Auswertung von Ausdrücken mit Konstanten

- **Symbolische Auswertung zur Entwurfszeit**

- Reduktion von Ausdrücken, deren “Wert” bestimmt werden kann
- Anwendbar, solange ein konstanter Teilausdruck vorhanden
- Formale Technik: Auffalten von Definitionen + Simplifikation
Je nach Inhalt der Wissensbank auch reine Simplifikation

Beispiel:

$ [x;5] \circ L $	unfold append
$\mapsto x . ([5] \circ L) $	unfold append
$\mapsto x . (5 . ([] \circ L)) $	unfold append
$\mapsto x . (5 . L) $	unfold length
$\mapsto 1 + 5 . L $	unfold length
$\mapsto 1 + 1 + L $	simplify
$\mapsto 2 + L $	

- **Benutzerinteraktion:**

- Auswahl von auszuwertendem Ausdruck und Optimierungstechnik

ENDLICHE DIFFERENZIERUNG

- **Inkrementelle Berechnung wiederkehrender Teilausdrücke**
 - Ersetze Teilausdruck $E[x]$ in $f(x)$ durch neue Variable c
 - Initialisiere mit c Ausdruck für Basiswerte
 - Bestimme differentielle Veränderungen bei rekursivem Aufruf
 - Effizienter als ständige Neuberechnung

ENDLICHE DIFFERENZIERUNG

- **Inkrementelle Berechnung wiederkehrender Teilausdrücke**

- Ersetze Teilausdruck $E[x]$ in $f(x)$ durch neue Variable c
- Initialisiere mit c Ausdruck für Basiswerte
- Bestimme differentielle Veränderungen bei rekursivem Aufruf
- Effizienter als ständige Neuberechnung

- **Endliche Differenzierung am Beispiel**

FUNCTION **Costas**_{aux}($n, s: \mathbb{Z} \times \text{Seq}(\mathbb{Z})$) WHERE ... RETURNS ...

$\equiv$ if $\{1..n\} \setminus \text{range}(s) = \emptyset$ then $\{s\}$ else $\emptyset$

$\cup \bigcup \{ \text{Costas}_{gs}(n, s \cdot i) \mid i \in \{1..n\} \setminus \text{range}(s) \wedge \forall j < |s|. s_{|s| - j} - i \notin \text{dtrow}(s, j) \}$

ENDLICHE DIFFERENZIERUNG

- **Inkrementelle Berechnung wiederkehrender Teilausdrücke**

- Ersetze Teilausdruck $E[x]$ in $f(x)$ durch neue Variable c
- Initialisiere mit c Ausdruck für Basiswerte
- Bestimme differentielle Veränderungen bei rekursivem Aufruf
- Effizienter als ständige Neuberechnung

- **Endliche Differenzierung am Beispiel**

```
FUNCTION Costasaux(n,s:ℤ×Seq(ℤ)) WHERE ... RETURNS ...  
≡      if {1..n}\range(s)=∅ then {s} else ∅  
      ∪ ⋃ {Costasgs(n,s·i) | i ∈ {1..n}\range(s) ∧ ∀j<|s|. s|s·i|-j-i ∉ dtrow(s,j)}
```

- Benutzer identifiziert $\{1..n\}\backslash\text{range}(s)$ als wiederkehrende Berechnung

ENDLICHE DIFFERENZIERUNG

- **Inkrementelle Berechnung wiederkehrender Teilausdrücke**

- Ersetze Teilausdruck $E[x]$ in $f(x)$ durch neue Variable c
- Initialisiere mit c Ausdruck für Basiswerte
- Bestimme differentielle Veränderungen bei rekursivem Aufruf
- Effizienter als ständige Neuberechnung

- **Endliche Differenzierung am Beispiel**

FUNCTION **Costas**_{aux}($n, s: \mathbb{Z} \times \text{Seq}(\mathbb{Z})$) WHERE ... RETURNS ...

$\equiv$ if $\{1..n\} \setminus \text{range}(s) = \emptyset$ then $\{s\}$ else $\emptyset$
 $\cup \bigcup \{ \text{Costas}_{gs}(n, s.i) \mid i \in \{1..n\} \setminus \text{range}(s) \wedge \forall j < |s|. s_{|s|.i-j} - i \notin \text{dtrow}(s, j) \}$

FUNCTION **Costas**_{aux}($n, s, \text{pool}: \mathbb{Z} \times \text{Seq}(\mathbb{Z}) \times \text{Seq}(\mathbb{Z})$) WHERE ... RETURNS ...

$\equiv$ if $\{1..n\} \setminus \text{range}(s) = \emptyset$ then $\{s\}$ else $\emptyset$
 $\cup \bigcup \{ \text{Costas}_{gs}(n, s.i, \{1..n\} \setminus \text{range}(s.i)) \mid i \in \{1..n\} \setminus \text{range}(s) \wedge \forall j < |s|. s_{|s|.i-j} - i \notin \text{dtrow}(s, j) \}$

- Benutzer identifiziert $\{1..n\} \setminus \text{range}(s)$ als wiederkehrende Berechnung
- System ändert **Costas**_{aux}(n, s) in **Costas**_{aux}(n, s, pool)
und ändert Aufrufe von **Costas**_{aux} entsprechend

ENDLICHE DIFFERENZIERUNG

● Inkrementelle Berechnung wiederkehrender Teilausdrücke

- Ersetze Teilausdruck $E[x]$ in $f(x)$ durch neue Variable c
- Initialisiere mit c Ausdruck für Basiswerte
- Bestimme differentielle Veränderungen bei rekursivem Aufruf
- Effizienter als ständige Neuberechnung

● Endliche Differenzierung am Beispiel

```
FUNCTION Costasaux(n,s:ℤ×Seq(ℤ)) WHERE ... RETURNS ...  
≡ if {1..n} \ range(s) = ∅ then {s} else ∅  
  ∪ ⋃ { Costasgs(n,s.i) | i ∈ {1..n} \ range(s) ∧ ∀j < |s|. s|s.i|-j - i ∉ dtrow(s,j) }
```

```
FUNCTION Costasaux(n,s,pool ...) WHERE ... pool = {1..n} \ range(s)  
≡ if {1..n} \ range(s) = ∅ then {s} else ∅  
  ∪ ⋃ { Costasgs(n,s.i,{1..n} \ range(s.i)) | i ∈ {1..n} \ range(s)  
        ∧ ∀j < |s|. s|s.i|-j - i ∉ dtrow(s,j) }
```

- Benutzer identifiziert $\{1..n\} \setminus \text{range}(s)$ als wiederkehrende Berechnung
- System ändert $\text{Costas}_{aux}(n,s)$ in $\text{Costas}_{aux}(n,s,pool)$
und ändert Aufrufe von Costas_{aux} entsprechend
- System ergänzt $pool = \{1..n\} \setminus \text{range}(s)$ zur Eingabebedingung

ENDLICHE DIFFERENZIERUNG

● Inkrementelle Berechnung wiederkehrender Teilausdrücke

- Ersetze Teilausdruck $E[x]$ in $f(x)$ durch neue Variable c
- Initialisiere mit c Ausdruck für Basiswerte
- Bestimme differentielle Veränderungen bei rekursivem Aufruf
- Effizienter als ständige Neuberechnung

● Endliche Differenzierung am Beispiel

FUNCTION $\text{Costas}_{aux}(n, s: \mathbb{Z} \times \text{Seq}(\mathbb{Z}))$ WHERE ... RETURNS ...
 $\equiv$ if $\{1..n\} \setminus \text{range}(s) = \emptyset$ then $\{s\}$ else $\emptyset$
 $\cup \bigcup \{ \text{Costas}_{gs}(n, s.i) \mid i \in \{1..n\} \setminus \text{range}(s) \wedge \forall j < |s|. s_{|s|.i-j} - i \notin \text{dtrow}(s, j) \}$

FUNCTION $\text{Costas}_{aux}(n, s, \text{pool} \dots)$ WHERE ... $\text{pool} = \{1..n\} \setminus \text{range}(s)$
 $\equiv$ if $\{1..n\} \setminus \text{range}(s) = \emptyset$ then $\{s\}$ else $\emptyset$
 $\cup \bigcup \{ \text{Costas}_{gs}(n, s.i, \{1..n\} \setminus \text{range}(s.i)) \mid i \in \{1..n\} \setminus \text{range}(s) \wedge \forall j < |s|. s_{|s|.i-j} - i \notin \text{dtrow}(s, j) \}$

- Benutzer identifiziert $\{1..n\} \setminus \text{range}(s)$ als wiederkehrende Berechnung
- System ändert $\text{Costas}_{aux}(n, s)$ in $\text{Costas}_{aux}(n, s, \text{pool})$
und ändert Aufrufe von Costas_{aux} entsprechend
- System ergänzt $\text{pool} = \{1..n\} \setminus \text{range}(s)$ zur Eingabebedingung
- System vereinfacht Vorkommen von $\{1..n\} \setminus \text{range}(s)$ zu pool

ENDLICHE DIFFERENZIERUNG

● Inkrementelle Berechnung wiederkehrender Teilausdrücke

- Ersetze Teilausdruck $E[x]$ in $f(x)$ durch neue Variable c
- Initialisiere mit c Ausdruck für Basiswerte
- Bestimme differentielle Veränderungen bei rekursivem Aufruf
- Effizienter als ständige Neuberechnung

● Endliche Differenzierung am Beispiel

FUNCTION $\text{Costas}_{aux}(n, s: \mathbb{Z} \times \text{Seq}(\mathbb{Z}))$ WHERE ... RETURNS ...
 $\equiv$ if $\{1..n\} \setminus \text{range}(s) = \emptyset$ then $\{s\}$ else $\emptyset$
 $\cup \bigcup \{ \text{Costas}_{gs}(n, s.i) \mid i \in \{1..n\} \setminus \text{range}(s) \wedge \forall j < |s|. s_{|s|.i-j} - i \notin \text{dtrow}(s, j) \}$

FUNCTION $\text{Costas}_{aux}(n, s, \text{pool} \dots)$ WHERE ... $\text{pool} = \{1..n\} \setminus \text{range}(s)$
 $\equiv$ if $\text{pool} = \emptyset$ then $\{s\}$ else $\emptyset$
 $\cup \bigcup \{ \text{Costas}_{gs}(n, s.i, \text{pool} \setminus \{i\}) \mid i \in \text{pool} \wedge \forall j < |s|. s_{|s|.i-j} - i \notin \text{dtrow}(s, j) \}$

- Benutzer identifiziert $\{1..n\} \setminus \text{range}(s)$ als wiederkehrende Berechnung
- System ändert $\text{Costas}_{aux}(n, s)$ in $\text{Costas}_{aux}(n, s, \text{pool})$
und ändert Aufrufe von Costas_{aux} entsprechend
- System ergänzt $\text{pool} = \{1..n\} \setminus \text{range}(s)$ zur Eingabebedingung
- System vereinfacht Vorkommen von $\{1..n\} \setminus \text{range}(s)$ zu pool

- **Abstraktion** über Ausdruck $E[x]$ in Funktion $f(x)$

FUNCTION *main*... WHERE ... RETURNS ... SUCH THAT... $\equiv$ $f(x_0)$

FUNCTION $f(x:D):R$ WHERE $I[x]$ RETURNS y SUCH THAT $O[x,y]$

$\equiv$ $E[x]$ $f(t[x])$..

- **Abstraktion** über Ausdruck $E[x]$ in Funktion $f(x)$

FUNCTION *main*... WHERE ... RETURNS ... SUCH THAT... $\equiv$ $f(x_0)$

FUNCTION $f(x:D):R$ WHERE $I[x]$ RETURNS y SUCH THAT $O[x,y]$
 $\equiv$ $E[x]$ $f(t[x])$..

– Erweitere f zu neuem f' , so daß $f(x) = f'(x, E[x])$

- **Abstraktion** über Ausdruck $E[x]$ in Funktion $f(x)$

FUNCTION *main*... WHERE ... RETURNS ... SUCH THAT... $\equiv$ $f(x_0)$

FUNCTION $f(x:D):R$ WHERE $I[x]$ RETURNS y SUCH THAT $O[x,y]$
 $\equiv$ $E[x]$ $f(t[x])$..

- Erweitere f zu neuem f' , so daß $f(x) = f'(x, E[x])$
- Ersetze alle Aufrufe der Art $f(t[x])$ durch $f'(t[x], E[t[x]])$

● **Abstraktion** über Ausdruck $E[x]$ in Funktion $f(x)$

FUNCTION *main*... WHERE ... RETURNS ... SUCH THAT... $\equiv$ $f(x_0)$

FUNCTION $f(x:D):R$ WHERE $I[x]$ RETURNS y SUCH THAT $O[x,y]$
 $\equiv$ $E[x]$ $f(t[x])$..

- Erweitere f zu neuem f' , so daß $f(x) = f'(x, E[x])$
- Ersetze alle Aufrufe der Art $f(t[x])$ durch $f'(t[x], E[t[x]])$
- Ergänze Gleichung $c=E[x]$ zur Eingabebedingung von f'

● Abstraktion über Ausdruck $E[x]$ in Funktion $f(x)$

FUNCTION *main*... WHERE ... RETURNS ... SUCH THAT... $\equiv$ $f(x_0)$

FUNCTION $f(x:D):R$ WHERE $I[x]$ RETURNS y SUCH THAT $O[x,y]$
 $\equiv$ $E[x]$ $f(t[x])$..

- Erweitere f zu neuem f' , so daß $f(x) = f'(x, E[x])$
 - Ersetze alle Aufrufe der Art $f(t[x])$ durch $f'(t[x], E[t[x]])$
 - Ergänze Gleichung $c=E[x]$ zur Eingabebedingung von f'
-

FUNCTION *main*... WHERE ... RETURNS ... SUCH THAT... $\equiv$ $f'(x_0, E[x_0])$

FUNCTION $f(x, c:D \times D'):R$ WHERE $I[x] \wedge c=E[x]$ RETURNS y SUCH THAT $O[x,y]$
 $\equiv$ $E[x]$ $f'(t[x], E[t[x]])$..

● Abstraktion über Ausdruck $E[x]$ in Funktion $f(x)$

FUNCTION *main*... WHERE ... RETURNS ... SUCH THAT... $\equiv$ $f(x_0)$

FUNCTION $f(x:D):R$ WHERE $I[x]$ RETURNS y SUCH THAT $O[x,y]$
 $\equiv$ $E[x]$ $f(t[x])$..

- Erweitere f zu neuem f' , so daß $f(x) = f'(x, E[x])$
 - Ersetze alle Aufrufe der Art $f(t[x])$ durch $f'(t[x], E[t[x]])$
 - Ergänze Gleichung $c=E[x]$ zur Eingabebedingung von f'
-

FUNCTION *main*... WHERE ... RETURNS ... SUCH THAT... $\equiv$ $f'(x_0, E[x_0])$

FUNCTION $f(x, c:D \times D'):R$ WHERE $I[x] \wedge c=E[x]$ RETURNS y SUCH THAT $O[x,y]$
 $\equiv$ $E[x]$ $f'(t[x], E[t[x]])$..

● Simplifikation mit Gleichung $c = E[x]$

- Transformiere Ausdrücke der Form $E[g[x]]$ in die Form $g'[E[x]]$
- Ersetze alle Vorkommen von $E[x]$ durch c

● **Abstraktion** über Ausdruck $E[x]$ in Funktion $f(x)$

FUNCTION *main*... WHERE ... RETURNS ... SUCH THAT... $\equiv$ $f(x_0)$

FUNCTION $f(x:D):R$ WHERE $I[x]$ RETURNS y SUCH THAT $O[x,y]$
 $\equiv$ $E[x]$ $f(t[x])$..

- Erweitere f zu neuem f' , so daß $f(x) = f'(x, E[x])$
 - Ersetze alle Aufrufe der Art $f(t[x])$ durch $f'(t[x], E[t[x]])$
 - Ergänze Gleichung $c=E[x]$ zur Eingabebedingung von f'
-

FUNCTION *main*... WHERE ... RETURNS ... SUCH THAT... $\equiv$ $f'(x_0, E[x_0])$

FUNCTION $f(x, c:D \times D'):R$ WHERE $I[x] \wedge c=E[x]$ RETURNS y SUCH THAT $O[x,y]$
 $\equiv$ $E[x]$ $f'(t[x], E[t[x]])$..

● **Simplifikation** mit Gleichung $c = E[x]$

- Transformiere Ausdrücke der Form $E[g[x]]$ in die Form $g'[E[x]]$
- Ersetze alle Vorkommen von $E[x]$ durch c

● **Benutzerinteraktion:**

- Auswahl des zu ersetzenden Teilausdrucks $E[x]$ in $f(x)$

ENDLICHE DIFFERENZIERUNG VON Costas_{aux}

FUNCTION Costas_{aux} ($n, s: \mathbb{Z} \times \text{Seq}(\mathbb{Z})$)
WHERE $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s, j))$
RETURNS $\{p: \text{Seq}(\mathbb{Z}) \mid \text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p, j))\}$
 $\equiv$ if $\{1..n\} \setminus \text{range}(s) = \emptyset$ then $\{s\}$ else $\emptyset$
 $\cup \bigcup \{ \text{Costas}_{gs}(n, s \cdot i) \mid i \in \{1..n\} \setminus \text{range}(s) \wedge \forall j < |s|. s_{|s| \cdot i - j} - i \notin \text{dtrow}(s, j) \}$

ENDLICHE DIFFERENZIERUNG VON Costas_{aux}

```
FUNCTION  $\text{Costas}_{aux}$  ( $n, s: \mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )  
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s, j))$   
  RETURNS  $\{p: \text{Seq}(\mathbb{Z}) \mid \text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p, j))\}$   
 $\equiv$     if  $\{1..n\} \setminus \text{range}(s) = \emptyset$  then  $\{s\}$  else  $\emptyset$   
     $\cup \bigcup \{ \text{Costas}_{gs}(n, s \cdot i) \mid i \in \{1..n\} \setminus \text{range}(s) \wedge \forall j < |s|. s_{|s| \cdot i - j} - i \notin \text{dtrow}(s, j) \}$ 
```

Benutzer identifiziert $\{1..n\} \setminus \text{range}(s)$ als wiederkehrende Berechnung

ENDLICHE DIFFERENZIERUNG VON Costas_{aux}

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$ 
  if  $\{1..n\} \setminus \text{range}(s) = \emptyset$  then  $\{s\}$  else  $\emptyset$ 
   $\cup \bigcup \{ \text{Costas}_{gs}(n,s \cdot i) \mid i \in \{1..n\} \setminus \text{range}(s) \wedge \forall j < |s|. s_{|s| \cdot i - j} - i \notin \text{dtrow}(s,j) \}$ 

```

```

FUNCTION  $\text{Costas}_{aux}$  (n,s,pool: $\mathbb{Z} \times \text{Seq}(\mathbb{Z}) \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
     $\wedge \text{pool} = \{1..n\} \setminus \text{range}(s)$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$ 
  if  $\text{pool} = \emptyset$  then  $\{s\}$  else  $\emptyset$ 
   $\cup \bigcup \{ \text{Costas}_{gs}(n,s \cdot i, \text{pool} \setminus \{i\}) \mid i \in \text{pool} \wedge \forall j < |s|. s_{|s| \cdot i - j} - i \notin \text{dtrow}(s,j) \}$ 

```

Benutzer identifiziert $\{1..n\} \setminus \text{range}(s)$ als wiederkehrende Berechnung
 System ändert $\text{Costas}_{aux}(n,s)$ in $\text{Costas}_{aux}(n,s,\text{pool})$ und ändert Aufrufe
 System ergänzt $\text{pool} = \{1..n\} \setminus \text{range}(s)$ zur Eingabebedingung
 und vereinfacht Vorkommen von $\{1..n\} \setminus \text{range}(s)$ zu pool

ENDLICHE DIFFERENZIERUNG VON Costas_{aux}

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$ 
  if  $\{1..n\} \setminus \text{range}(s) = \emptyset$  then  $\{s\}$  else  $\emptyset$ 
   $\cup \bigcup \{ \text{Costas}_{gs}(n,s \cdot i) \mid i \in \{1..n\} \setminus \text{range}(s) \wedge \forall j < |s|. s_{|s| \cdot i - j} - i \notin \text{dtrow}(s,j) \}$ 

```

```

FUNCTION  $\text{Costas}_{aux}$  (n,s,pool: $\mathbb{Z} \times \text{Seq}(\mathbb{Z}) \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
   $\wedge \text{pool} = \{1..n\} \setminus \text{range}(s)$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$ 
  if  $\text{pool} = \emptyset$  then  $\{s\}$  else  $\emptyset$ 
   $\cup \bigcup \{ \text{Costas}_{gs}(n,s \cdot i, \text{pool} \setminus \{i\}) \mid i \in \text{pool} \wedge \forall j < |s|. s_{|s| \cdot i - j} - i \notin \text{dtrow}(s,j) \}$ 

```

Benutzer identifiziert $|s|$ als wiederkehrende Berechnung

ENDLICHE DIFFERENZIERUNG VON Costas_{aux}

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$ 
  if  $\{1..n\} \setminus \text{range}(s) = \emptyset$  then  $\{s\}$  else  $\emptyset$ 
   $\cup \bigcup \{ \text{Costas}_{gs}(n,s \cdot i) \mid i \in \{1..n\} \setminus \text{range}(s) \wedge \forall j < |s|. s_{|s|-j} - i \notin \text{dtrow}(s,j) \}$ 

```

```

FUNCTION  $\text{Costas}_{aux}$  (n,s,pool,ssize: $\mathbb{Z} \times \text{Seq}(\mathbb{Z}) \times \text{Seq}(\mathbb{Z}) \times \mathbb{Z}$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
   $\wedge \text{pool} = \{1..n\} \setminus \text{range}(s) \wedge \text{ssize} = |s|$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$ 
  if  $\text{pool} = \emptyset$  then  $\{s\}$  else  $\emptyset$ 
   $\cup \bigcup \{ \text{Costas}_{gs}(n,s \cdot i, \text{pool} \setminus \{i\}, \text{ssize}+1) \mid i \in \text{pool}$ 
   $\wedge \forall j < \text{ssize}. s_{\text{ssize}+1-j} - i \notin \text{dtrow}(s,j) \}$ 

```

Benutzer identifiziert $|s|$ als wiederkehrende Berechnung

System ändert $\text{Costas}_{aux}(n,s,\text{pool})$ in $\text{Costas}_{aux}(n,s,\text{pool},\text{ssize})$

System ergänzt $\text{ssize} = |s|$ zur Eingabebedingung

und vereinfacht Vorkommen von $|s|$ zu ssize

ENDLICHE DIFFERENZIERUNG VON Costas_{aux}

```

FUNCTION  $\text{Costas}_{aux}$  (n,s: $\mathbb{Z} \times \text{Seq}(\mathbb{Z})$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$ 
  if  $\{1..n\} \setminus \text{range}(s) = \emptyset$  then  $\{s\}$  else  $\emptyset$ 
   $\cup \bigcup \{ \text{Costas}_{gs}(n,s.i) \mid i \in \{1..n\} \setminus \text{range}(s) \wedge \forall j < |s|. s_{|s|.i-j} - i \notin \text{dtrow}(s,j) \}$ 

```

```

FUNCTION  $\text{Costas}_{aux}$  (n,s,pool,ssize: $\mathbb{Z} \times \text{Seq}(\mathbb{Z}) \times \text{Seq}(\mathbb{Z}) \times \mathbb{Z}$ )
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s,j))$ 
   $\wedge \text{pool} = \{1..n\} \setminus \text{range}(s) \wedge \text{ssize} = |s|$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$ 
  if  $\text{pool} = \emptyset$  then  $\{s\}$  else  $\emptyset$ 
   $\cup \bigcup \{ \text{Costas}_{gs}(n,s.i,\text{pool} \setminus \{i\},\text{ssize}+1) \mid i \in \text{pool}$ 
   $\wedge \forall j < \text{ssize}. s_{\text{ssize}+1-j} - i \notin \text{dtrow}(s,j) \}$ 

```

Modifizierter Aufruf aus Hauptfunktion

```

FUNCTION  $\text{Costas}$  (n: $\mathbb{Z}$ ) WHERE  $n \geq 1$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv \text{Costas}_{gs}(n, [], \{1..n\}, 0)$ 

```

- **Separate Analyse von Einzelfällen**
 - Auswertung von Tests aus anderen Programmteilen
 - Zweck: globale Vereinfachung durch lokale Einzelanalyse
 - Formal: Erzeugung eines Kontexts + CD-Simplifikation

- **Separate Analyse von Einzelfällen**

- Auswertung von Tests aus anderen Programmteilen
- Zweck: globale Vereinfachung durch lokale Einzelanalyse
- Formal: Erzeugung eines Kontexts + CD-Simplifikation

- **Kontexterzeugung mit Prädikat P**

- Ersetze Ausdruck $E[x]$ durch `if  $P[x]$  then  $E[x]$  else  $E[x]$`
- Vereinfache $E[x]$ in den entsprechenden Kontexten $P[x]$ und $\neg P[x]$
- Distribuiere `if  $P[x]$  then...else...`
über Ausdrücke außerhalb von $E[x]$

- **Separate Analyse von Einzelfällen**

- Auswertung von Tests aus anderen Programmteilen
- Zweck: globale Vereinfachung durch lokale Einzelanalyse
- Formal: Erzeugung eines Kontexts + CD-Simplifikation

- **Kontexterzeugung mit Prädikat P**

- Ersetze Ausdruck $E[x]$ durch `if  $P[x]$  then  $E[x]$  else  $E[x]$`
- Vereinfache $E[x]$ in den entsprechenden Kontexten $P[x]$ und $\neg P[x]$
- Distributiere `if  $P[x]$  then...else...`
über Ausdrücke außerhalb von $E[x]$

- **Benutzerinteraktion:**

- Auswahl des zu ersetzenden Teilausdrucks $E[x]$
- Auswahl des Prädikats $P[x]$
- Auslösung der nachfolgenden Simplifikationen

FALLANALYSE IN Costas_{aux}

```

FUNCTION  $\text{Costas}_{aux}(n, s, \text{pool}, \text{ssize} : \mathbb{Z} \times \text{Seq}(\mathbb{Z}) \times \text{Seq}(\mathbb{Z}) \times \mathbb{Z})$ 
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s, j))$ 
         $\wedge \text{pool} = \{1..n\} \setminus \text{range}(s) \wedge \text{ssize} = |s|$ 
  RETURNS  $\{p : \text{Seq}(\mathbb{Z}) \mid \text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p, j))\}$ 
 $\equiv$     if  $\text{pool} = \emptyset$  then  $\{s\}$  else  $\emptyset$ 
 $\cup \bigcup \{ \text{Costas}_{gs}(n, s \cdot i, \text{pool} \setminus \{i\}, \text{ssize} + 1) \mid i \in \text{pool}$ 
                                                 $\wedge \forall j < \text{ssize}. s_{\text{ssize} + 1 - j} - i \notin \text{dtrow}(s, j) \}$ 

```

FALLANALYSE IN Costas_{aux}

```
FUNCTION  $\text{Costas}_{aux}(n, s, \text{pool}, \text{ssize} : \mathbb{Z} \times \text{Seq}(\mathbb{Z}) \times \text{Seq}(\mathbb{Z}) \times \mathbb{Z})$   
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s, j))$   
         $\wedge \text{pool} = \{1..n\} \setminus \text{range}(s) \wedge \text{ssize} = |s|$   
  RETURNS  $\{p : \text{Seq}(\mathbb{Z}) \mid \text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p, j))\}$   
 $\equiv$     if  $\text{pool} = \emptyset$  then  $\{s\}$  else  $\emptyset$   
     $\cup \bigcup \{ \text{Costas}_{gs}(n, s \cdot i, \text{pool} \setminus \{i\}, \text{ssize} + 1) \mid i \in \text{pool}$   
                                               $\wedge \forall j < \text{ssize}. s_{\text{ssize}+1-j} - i \notin \text{dtrow}(s, j) \}$ 
```

Markierung von Ausdruck und Testprädikat

FALLANALYSE IN Costas_{aux}

```

FUNCTION  $\text{Costas}_{aux}(n, s, \text{pool}, \text{ssize} : \mathbb{Z} \times \text{Seq}(\mathbb{Z}) \times \text{Seq}(\mathbb{Z}) \times \mathbb{Z})$ 
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s, j))$ 
         $\wedge \text{pool} = \{1..n\} \setminus \text{range}(s) \wedge \text{ssize} = |s|$ 
  RETURNS  $\{p : \text{Seq}(\mathbb{Z}) \mid \text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p, j))\}$ 
 $\equiv$    if  $\text{pool} = \emptyset$  then  $\{s\}$  else  $\emptyset$ 
       $\cup \bigcup \{ \text{Costas}_{gs}(n, s \cdot i, \text{pool} \setminus \{i\}, \text{ssize} + 1) \mid i \in \text{pool}$ 
                                                 $\wedge \forall j < \text{ssize}. s_{\text{ssize}+1-j} - i \notin \text{dtrow}(s, j) \}$ 

```

```

FUNCTION  $\text{Costas}_{aux}(n, s, \text{pool}, \text{ssize} : \mathbb{Z} \times \text{Seq}(\mathbb{Z}) \times \text{Seq}(\mathbb{Z}) \times \mathbb{Z})$ 
  WHERE ... RETURNS ...
 $\equiv$    if  $\text{pool} = \emptyset$  then  $\{s\}$  else  $\emptyset$ 
       $\cup$  if  $\text{pool} = \emptyset$  then  $\bigcup \{ \text{Costas}_{gs}(n, s \cdot i, \text{pool} \setminus \{i\}, \text{ssize} + 1) \mid i \in \text{pool}$ 
                                                 $\wedge \forall j < \text{ssize}. s_{\text{ssize}+1-j} - i \notin \text{dtrow}(s, j) \}$ 
        else  $\bigcup \{ \text{Costas}_{gs}(n, s \cdot i, \text{pool} \setminus \{i\}, \text{ssize} + 1) \mid i \in \text{pool}$ 
                                                 $\wedge \forall j < \text{ssize}. s_{\text{ssize}+1-j} - i \notin \text{dtrow}(s, j) \}$ 

```

Aufspalten des Ausdrucks auf beide Fälle

FALLANALYSE IN Costas_{aux}

```

FUNCTION  $\text{Costas}_{aux}(n,s,pool,ssize:\mathbb{Z}\times\text{Seq}(\mathbb{Z})\times\text{Seq}(\mathbb{Z})\times\mathbb{Z})$ 
  WHERE  $n\geq 1 \wedge \text{range}(s)\subseteq\{1..n\} \wedge \text{nodups}(s) \wedge \forall j<|s|. \text{nodups}(\text{dtrow}(s,j))$ 
         $\wedge \text{pool} = \{1..n\}\setminus\text{range}(s) \wedge \text{ssize}=|s|$ 
  RETURNS  $\{p:\text{Seq}(\mathbb{Z}) \mid \text{perm}(p,\{1..n\}) \wedge s\sqsubseteq p \wedge \forall j<|p|. \text{nodups}(\text{dtrow}(p,j))\}$ 
 $\equiv$    if  $\text{pool}=\emptyset$  then  $\{s\}$  else  $\emptyset$ 
         $\cup \bigcup \{ \text{Costas}_{gs}(n,s\cdot i, \text{pool}\setminus\{i\}, \text{ssize}+1) \mid i\in\text{pool}$ 
                                      $\wedge \forall j<\text{ssize}. s_{\text{ssize}+1-j} - i \notin \text{dtrow}(s,j) \}$ 

```

```

FUNCTION  $\text{Costas}_{aux}(n,s,pool,ssize:\mathbb{Z}\times\text{Seq}(\mathbb{Z})\times\text{Seq}(\mathbb{Z})\times\mathbb{Z})$ 
  WHERE ... RETURNS ...
 $\equiv$    if  $\text{pool}=\emptyset$  then  $\{s\}$  else  $\emptyset$ 
         $\cup$  if  $\text{pool}=\emptyset$  then  $\bigcup \{ \text{Costas}_{gs}(n,s\cdot i, \text{pool}\setminus\{i\}, \text{ssize}+1) \mid i\in\text{pool}$ 
                                      $\wedge \forall j<\text{ssize}. s_{\text{ssize}+1-j} - i \notin \text{dtrow}(s,j) \}$ 
        else  $\bigcup \{ \text{Costas}_{gs}(n,s\cdot i, \text{pool}\setminus\{i\}, \text{ssize}+1) \mid i\in\text{pool}$ 
                                      $\wedge \forall j<\text{ssize}. s_{\text{ssize}+1-j} - i \notin \text{dtrow}(s,j) \}$ 

```

Domänenwissen: $\bigcup \{ f(i) \mid i\in\emptyset \} = \emptyset$

Kontext der Fallanalyse: $\text{pool}=\emptyset$

FALLANALYSE IN Costas_{aux}

```

FUNCTION  $\text{Costas}_{aux}(n, s, \text{pool}, \text{ssize} : \mathbb{Z} \times \text{Seq}(\mathbb{Z}) \times \text{Seq}(\mathbb{Z}) \times \mathbb{Z})$ 
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s, j))$ 
         $\wedge \text{pool} = \{1..n\} \setminus \text{range}(s) \wedge \text{ssize} = |s|$ 
  RETURNS  $\{p : \text{Seq}(\mathbb{Z}) \mid \text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p, j))\}$ 
 $\equiv$    if  $\text{pool} = \emptyset$  then  $\{s\}$  else  $\emptyset$ 
       $\cup \bigcup \{ \text{Costas}_{gs}(n, s \cdot i, \text{pool} \setminus \{i\}, \text{ssize}+1) \mid i \in \text{pool}$ 
                                                 $\wedge \forall j < \text{ssize}. s_{\text{ssize}+1-j} - i \notin \text{dtrow}(s, j) \}$ 

```

```

FUNCTION  $\text{Costas}_{aux}(n, s, \text{pool}, \text{ssize} : \mathbb{Z} \times \text{Seq}(\mathbb{Z}) \times \text{Seq}(\mathbb{Z}) \times \mathbb{Z})$ 
  WHERE ... RETURNS ...
 $\equiv$    if  $\text{pool} = \emptyset$  then  $\{s\}$  else  $\emptyset$ 
       $\cup$  if  $\text{pool} = \emptyset$  then  $\emptyset$ 
          else  $\bigcup \{ \text{Costas}_{gs}(n, s \cdot i, \text{pool} \setminus \{i\}, \text{ssize}+1) \mid i \in \text{pool}$ 
                                                         $\wedge \forall j < \text{ssize}. s_{\text{ssize}+1-j} - i \notin \text{dtrow}(s, j) \}$ 

```

Domänenwissen: $\bigcup \{ f(i) \mid i \in \emptyset \} = \emptyset$

Kontext der Fallanalyse: $\text{pool} = \emptyset$

FALLANALYSE IN Costas_{aux}

```

FUNCTION  $\text{Costas}_{aux}(n, s, \text{pool}, \text{ssize} : \mathbb{Z} \times \text{Seq}(\mathbb{Z}) \times \text{Seq}(\mathbb{Z}) \times \mathbb{Z})$ 
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s, j))$ 
         $\wedge \text{pool} = \{1..n\} \setminus \text{range}(s) \wedge \text{ssize} = |s|$ 
  RETURNS  $\{p : \text{Seq}(\mathbb{Z}) \mid \text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p, j))\}$ 
 $\equiv$    if  $\text{pool} = \emptyset$  then  $\{s\}$  else  $\emptyset$ 
       $\cup \bigcup \{ \text{Costas}_{gs}(n, s \cdot i, \text{pool} \setminus \{i\}, \text{ssize} + 1) \mid i \in \text{pool}$ 
                                                 $\wedge \forall j < \text{ssize}. s_{\text{ssize}+1-j} - i \notin \text{dtrow}(s, j) \}$ 

```

```

FUNCTION  $\text{Costas}_{aux}(n, s, \text{pool}, \text{ssize} : \mathbb{Z} \times \text{Seq}(\mathbb{Z}) \times \text{Seq}(\mathbb{Z}) \times \mathbb{Z})$ 
  WHERE ... RETURNS ...
 $\equiv$    if  $\text{pool} = \emptyset$  then  $\{s\}$  else  $\emptyset$ 
       $\cup$  if  $\text{pool} = \emptyset$  then  $\emptyset$ 
        else  $\bigcup \{ \text{Costas}_{gs}(n, s \cdot i, \text{pool} \setminus \{i\}, \text{ssize} + 1) \mid i \in \text{pool}$ 
                                                               $\wedge \forall j < \text{ssize}. s_{\text{ssize}+1-j} - i \notin \text{dtrow}(s, j) \}$ 

```

Distribution von $\text{if } \text{pool} = \emptyset \text{ then } \dots \text{ else } \dots$ über $\cup$

FALLANALYSE IN Costas_{aux}

```

FUNCTION  $\text{Costas}_{aux}(n, s, \text{pool}, \text{ssize} : \mathbb{Z} \times \text{Seq}(\mathbb{Z}) \times \text{Seq}(\mathbb{Z}) \times \mathbb{Z})$ 
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s, j))$ 
         $\wedge \text{pool} = \{1..n\} \setminus \text{range}(s) \wedge \text{ssize} = |s|$ 
  RETURNS  $\{p : \text{Seq}(\mathbb{Z}) \mid \text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p, j))\}$ 
 $\equiv$    if  $\text{pool} = \emptyset$  then  $\{s\}$  else  $\emptyset$ 
       $\cup \bigcup \{ \text{Costas}_{gs}(n, s \cdot i, \text{pool} \setminus \{i\}, \text{ssize} + 1) \mid i \in \text{pool}$ 
                                                 $\wedge \forall j < \text{ssize}. s_{\text{ssize}+1-j} - i \notin \text{dtrow}(s, j) \}$ 

```

```

FUNCTION  $\text{Costas}_{aux}(n, s, \text{pool}, \text{ssize} : \mathbb{Z} \times \text{Seq}(\mathbb{Z}) \times \text{Seq}(\mathbb{Z}) \times \mathbb{Z})$ 
  WHERE ... RETURNS ...
 $\equiv$    if  $\text{pool} = \emptyset$  then  $\{s\} \cup \emptyset$ 
        else  $\emptyset \cup \bigcup \{ \text{Costas}_{gs}(n, s \cdot i, \text{pool} \setminus \{i\}, \text{ssize} + 1) \mid i \in \text{pool}$ 
                                                 $\wedge \forall j < \text{ssize}. s_{\text{ssize}+1-j} - i \notin \text{dtrow}(s, j) \}$ 

```

Distribution von $\text{if } \text{pool} = \emptyset \text{ then } \dots \text{ else } \dots$ über $\bigcup$

FALLANALYSE IN Costas_{aux}

```

FUNCTION  $\text{Costas}_{aux}(n, s, \text{pool}, \text{ssize} : \mathbb{Z} \times \text{Seq}(\mathbb{Z}) \times \text{Seq}(\mathbb{Z}) \times \mathbb{Z})$ 
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s, j))$ 
         $\wedge \text{pool} = \{1..n\} \setminus \text{range}(s) \wedge \text{ssize} = |s|$ 
  RETURNS  $\{p : \text{Seq}(\mathbb{Z}) \mid \text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p, j))\}$ 
 $\equiv$    if  $\text{pool} = \emptyset$  then  $\{s\}$  else  $\emptyset$ 
       $\cup \bigcup \{ \text{Costas}_{gs}(n, s \cdot i, \text{pool} \setminus \{i\}, \text{ssize} + 1) \mid i \in \text{pool}$ 
                                      $\wedge \forall j < \text{ssize}. s_{\text{ssize}+1-j} - i \notin \text{dtrow}(s, j) \}$ 

```

```

FUNCTION  $\text{Costas}_{aux}(n, s, \text{pool}, \text{ssize} : \mathbb{Z} \times \text{Seq}(\mathbb{Z}) \times \text{Seq}(\mathbb{Z}) \times \mathbb{Z})$ 
  WHERE ... RETURNS ...
 $\equiv$    if  $\text{pool} = \emptyset$  then  $\{s\} \cup \emptyset$ 
        else  $\emptyset \cup \bigcup \{ \text{Costas}_{gs}(n, s \cdot i, \text{pool} \setminus \{i\}, \text{ssize} + 1) \mid i \in \text{pool}$ 
                                      $\wedge \forall j < \text{ssize}. s_{\text{ssize}+1-j} - i \notin \text{dtrow}(s, j) \}$ 

```

Domänenwissen: $S \cup \emptyset = \emptyset = \emptyset \cup S$

FALLANALYSE IN Costas_{aux}

```

FUNCTION  $\text{Costas}_{aux}(n, s, \text{pool}, \text{ssize} : \mathbb{Z} \times \text{Seq}(\mathbb{Z}) \times \text{Seq}(\mathbb{Z}) \times \mathbb{Z})$ 
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s, j))$ 
         $\wedge \text{pool} = \{1..n\} \setminus \text{range}(s) \wedge \text{ssize} = |s|$ 
  RETURNS  $\{p : \text{Seq}(\mathbb{Z}) \mid \text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p, j))\}$ 
 $\equiv$    if  $\text{pool} = \emptyset$  then  $\{s\}$  else  $\emptyset$ 
       $\cup \bigcup \{ \text{Costas}_{gs}(n, s \cdot i, \text{pool} \setminus \{i\}, \text{ssize}+1) \mid i \in \text{pool}$ 
                                      $\wedge \forall j < \text{ssize}. s_{\text{ssize}+1-j} - i \notin \text{dtrow}(s, j) \}$ 

```

```

FUNCTION  $\text{Costas}_{aux}(n, s, \text{pool}, \text{ssize} : \mathbb{Z} \times \text{Seq}(\mathbb{Z}) \times \text{Seq}(\mathbb{Z}) \times \mathbb{Z})$ 
  WHERE ... RETURNS ...
 $\equiv$    if  $\text{pool} = \emptyset$  then  $\{s\}$ 
        else  $\bigcup \{ \text{Costas}_{gs}(n, s \cdot i, \text{pool} \setminus \{i\}, \text{ssize}+1) \mid i \in \text{pool}$ 
                                      $\wedge \forall j < \text{ssize}. s_{\text{ssize}+1-j} - i \notin \text{dtrow}(s, j) \}$ 

```

Domänenwissen: $S \cup \emptyset = \emptyset = \emptyset \cup S$

FALLANALYSE IN Costas_{aux}

```

FUNCTION  $\text{Costas}_{aux}(n, s, \text{pool}, \text{ssize} : \mathbb{Z} \times \text{Seq}(\mathbb{Z}) \times \text{Seq}(\mathbb{Z}) \times \mathbb{Z})$ 
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s, j))$ 
         $\wedge \text{pool} = \{1..n\} \setminus \text{range}(s) \wedge \text{ssize} = |s|$ 
  RETURNS  $\{p : \text{Seq}(\mathbb{Z}) \mid \text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p, j))\}$ 
 $\equiv$    if  $\text{pool} = \emptyset$  then  $\{s\}$  else  $\emptyset$ 
       $\cup \bigcup \{ \text{Costas}_{gs}(n, s \cdot i, \text{pool} \setminus \{i\}, \text{ssize} + 1) \mid i \in \text{pool}$ 
                                                 $\wedge \forall j < \text{ssize}. s_{\text{ssize}+1-j} - i \notin \text{dtrow}(s, j) \}$ 

```

```

FUNCTION  $\text{Costas}_{aux}(n, s, \text{pool}, \text{ssize} : \mathbb{Z} \times \text{Seq}(\mathbb{Z}) \times \text{Seq}(\mathbb{Z}) \times \mathbb{Z})$ 
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s, j))$ 
         $\wedge \text{pool} = \{1..n\} \setminus \text{range}(s) \wedge \text{ssize} = |s|$ 
  RETURNS  $\{p : \text{Seq}(\mathbb{Z}) \mid \text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p, j))\}$ 
 $\equiv$    if  $\text{pool} = \emptyset$  then  $\{s\}$ 
      else  $\bigcup \{ \text{Costas}_{gs}(n, s \cdot i, \text{pool} \setminus \{i\}, \text{ssize} + 1) \mid i \in \text{pool}$ 
                                                 $\wedge \forall j < \text{ssize}. s_{\text{ssize}+1-j} - i \notin \text{dtrow}(s, j) \}$ 

```

Ersetze abstrakte Datentypen durch effiziente konkrete Implementierung

- **Endliche Mengen**

- **Listen**: Standardimplementierung
- **Bitvektor**: Mengen über endlichem Domain
- **Charakteristische Funktion**: effiziente Elementrelation/Einfügen/Löschen

Ersetze abstrakte Datentypen durch effiziente konkrete Implementierung

● Endliche Mengen

- **Listen**: Standardimplementierung
- **Bitvektor**: Mengen über endlichem Domain
- **Charakteristische Funktion**: effiziente Elementrelation/Einfügen/Löschen

● Folgen

- **Verkettete Liste**: Standardimplementierung
- **Umgekehrt verkettete Liste**: gut für append-Operation •

Ersetze abstrakte Datentypen durch effiziente konkrete Implementierung

● Endliche Mengen

- **Listen**: Standardimplementierung
- **Bitvektor**: Mengen über endlichem Domain
- **Charakteristische Funktion**: effiziente Elementrelation/Einfügen/Löschen

● Folgen

- **Verkettete Liste**: Standardimplementierung
- **Umgekehrt verkettete Liste**: gut für append-Operation •

● Benutzerinteraktion:

- System stellt **Auswahl von Implementierungen** bereit
- Benutzer wählt Nichtstandard-Implementierung **einzeln für jede Variable**
- System ersetzt abstrakte Notation durch konkrete Implementierung und fügt ggf. Konversionen ein

DATENTYPVERFEINERUNG FÜR Costas_{aux}

```
FUNCTION  $\text{Costas}_{aux}(n, s, \text{pool}, \text{ssize} : \mathbb{Z} \times \text{Seq}(\mathbb{Z}) \times \text{Seq}(\mathbb{Z}) \times \mathbb{Z})$   
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s, j))$   
         $\wedge \text{pool} = \{1..n\} \setminus \text{range}(s) \wedge \text{ssize} = |s|$   
  RETURNS  $\{p : \text{Seq}(\mathbb{Z}) \mid \text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p, j))\}$   
 $\equiv$    if  $\text{pool} = \emptyset$  then  $\{s\}$   
        else  $\bigcup \{ \text{Costas}_{gs}(n, s \cdot i, \text{pool} \setminus \{i\}, \text{ssize} + 1) \mid i \in \text{pool}$   
                 $\wedge \forall j < \text{ssize}. s_{\text{ssize} + 1 - j} - i \notin \text{dtrow}(s, j) \}$ 
```

DATENTYPVERFEINERUNG FÜR Costas_{aux}

```
FUNCTION  $\text{Costas}_{aux}(\mathbf{n}, s, \text{pool}, \text{ssize} : \mathbb{Z} \times \text{Seq}(\mathbb{Z}) \times \text{Seq}(\mathbb{Z}) \times \mathbb{Z})$   
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s, j))$   
         $\wedge \text{pool} = \{1..n\} \setminus \text{range}(s) \wedge \text{ssize} = |s|$   
  RETURNS  $\{p : \text{Seq}(\mathbb{Z}) \mid \text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p, j))\}$   
 $\equiv$    if  $\text{pool} = \emptyset$  then  $\{s\}$   
        else  $\bigcup \{ \text{Costas}_{gs}(\mathbf{n}, s \cdot i, \text{pool} \setminus \{i\}, \text{ssize} + 1) \mid i \in \text{pool}$   
                 $\wedge \forall j < \text{ssize}. s_{\text{ssize}+1-j} - i \notin \text{dtrow}(s, j) \}$ 
```

$\mathbf{n} : \mathbb{Z}$

$\mapsto$ Standardimplementierung positiver ganzen Zahlen

DATENTYPVERFEINERUNG FÜR Costas_{aux}

```
FUNCTION  $\text{Costas}_{aux}(n, \mathbf{s}, \text{pool}, \text{ssize} : \mathbb{Z} \times \text{Seq}(\mathbb{Z}) \times \text{Seq}(\mathbb{Z}) \times \mathbb{Z})$   
  WHERE  $n \geq 1 \wedge \text{range}(\mathbf{s}) \subseteq \{1..n\} \wedge \text{nodups}(\mathbf{s}) \wedge \forall j < |\mathbf{s}|. \text{nodups}(\text{dtrow}(\mathbf{s}, j))$   
         $\wedge \text{pool} = \{1..n\} \setminus \text{range}(\mathbf{s}) \wedge \text{ssize} = |\mathbf{s}|$   
  RETURNS  $\{p : \text{Seq}(\mathbb{Z}) \mid \text{perm}(p, \{1..n\}) \wedge \mathbf{s} \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p, j))\}$   
 $\equiv$    if  $\text{pool} = \emptyset$  then  $\{\mathbf{s}\}$   
        else  $\bigcup \{ \text{Costas}_{gs}(n, \mathbf{s} \cdot i, \text{pool} \setminus \{i\}, \text{ssize} + 1) \mid i \in \text{pool}$   
                 $\wedge \forall j < \text{ssize}. \mathbf{s}_{\text{ssize}+1-j} - i \notin \text{dtrow}(\mathbf{s}, j) \}$ 
```

$n : \mathbb{Z}$ $\mapsto$ Standardimplementierung positiver ganzen Zahlen

$\mathbf{s} : \text{Seq}(\mathbb{Z})$, Elemente werden hinten angehängt $\mapsto$ umgekehrt verkettete Liste

DATENTYPVERFEINERUNG FÜR Costas_{aux}

```
FUNCTION  $\text{Costas}_{aux}(n, s, \text{pool}, \text{ssize} : \mathbb{Z} \times \text{Seq}(\mathbb{Z}) \times \text{Seq}(\mathbb{Z}) \times \mathbb{Z})$   
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s, j))$   
         $\wedge \text{pool} = \{1..n\} \setminus \text{range}(s) \wedge \text{ssize} = |s|$   
  RETURNS  $\{p : \text{Seq}(\mathbb{Z}) \mid \text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p, j))\}$   
 $\equiv$    if  $\text{pool} = \emptyset$  then  $\{s\}$   
        else  $\bigcup \{ \text{Costas}_{gs}(n, s \cdot i, \text{pool} \setminus \{i\}, \text{ssize} + 1) \mid i \in \text{pool}$   
                 $\wedge \forall j < \text{ssize}. s_{\text{ssize}+1-j} - i \notin \text{dtrow}(s, j) \}$ 
```

$n : \mathbb{Z}$ $\mapsto$ Standardimplementierung positiver ganzen Zahlen
 $s : \text{Seq}(\mathbb{Z})$, Elemente werden hinten angehängt $\mapsto$ umgekehrt verkettete Liste
 $\text{pool} : \text{Set}(\mathbb{Z})$: Elemente werden aus fester Menge entnommen $\mapsto$ Bitvektor

DATENTYPVERFEINERUNG FÜR Costas_{aux}

```
FUNCTION  $\text{Costas}_{aux}(n, s, \text{pool}, \text{ssize} : \mathbb{Z} \times \text{Seq}(\mathbb{Z}) \times \text{Seq}(\mathbb{Z}) \times \mathbb{Z})$   
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s, j))$   
         $\wedge \text{pool} = \{1..n\} \setminus \text{range}(s) \wedge \text{ssize} = |s|$   
  RETURNS  $\{p : \text{Seq}(\mathbb{Z}) \mid \text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p, j))\}$   
 $\equiv$  if  $\text{pool} = \emptyset$  then  $\{s\}$   
    else  $\bigcup \{ \text{Costas}_{gs}(n, s \cdot i, \text{pool} \setminus \{i\}, \text{ssize}+1) \mid i \in \text{pool}$   
               $\wedge \forall j < \text{ssize}. s_{\text{ssize}+1-j} - i \notin \text{dtrow}(s, j) \}$ 
```

$n : \mathbb{Z}$ $\mapsto$ Standardimplementierung positiver ganzen Zahlen
 $s : \text{Seq}(\mathbb{Z})$, Elemente werden hinten angehängt $\mapsto$ umgekehrt verkettete Liste
 $\text{pool} : \text{Set}(\mathbb{Z})$: Elemente werden aus fester Menge entnommen $\mapsto$ Bitvektor
 $\text{ssize} : \mathbb{Z}$ $\mapsto$ Standardimplementierung positiver ganzen Zahlen

DATENTYPVERFEINERUNG FÜR Costas_{aux}

```
FUNCTION  $\text{Costas}_{aux}(n, s, \text{pool}, \text{ssize} : \mathbb{Z} \times \text{Seq}(\mathbb{Z}) \times \text{Seq}(\mathbb{Z}) \times \mathbb{Z})$   
  WHERE  $n \geq 1 \wedge \text{range}(s) \subseteq \{1..n\} \wedge \text{nodups}(s) \wedge \forall j < |s|. \text{nodups}(\text{dtrow}(s, j))$   
         $\wedge \text{pool} = \{1..n\} \setminus \text{range}(s) \wedge \text{ssize} = |s|$   
  RETURNS  $\{p : \text{Seq}(\mathbb{Z}) \mid \text{perm}(p, \{1..n\}) \wedge s \sqsubseteq p \wedge \forall j < |p|. \text{nodups}(\text{dtrow}(p, j))\}$   
 $\equiv$  if  $\text{pool} = \emptyset$  then  $\{s\}$   
    else  $\bigcup \{ \text{Costas}_{gs}(n, s \cdot i, \text{pool} \setminus \{i\}, \text{ssize} + 1) \mid i \in \text{pool}$   
               $\wedge \forall j < \text{ssize}. s_{\text{ssize}+1-j} - i \notin \text{dtrow}(s, j) \}$ 
```

$n : \mathbb{Z}$ $\mapsto$ Standardimplementierung positiver ganzen Zahlen

$s : \text{Seq}(\mathbb{Z})$, Elemente werden hinten angehängt $\mapsto$ umgekehrt verkettete Liste

$\text{pool} : \text{Set}(\mathbb{Z})$: Elemente werden aus fester Menge entnommen $\mapsto$ Bitvektor

$\text{ssize} : \mathbb{Z}$ $\mapsto$ Standardimplementierung positiver ganzen Zahlen