GOETHE UNIVERSITÄT FRANKFURT AM MAIN

www.uni-frankfurt.de

Programmierparadigmen und Compilerbau (PPDC)

3. Weitere Beispiele zur Unifikation

Sommersemester 2021

PD Dr. Arne Nägel

*Basierend auf Unterlagen von Prof. Dr. Manfred Schmidt-Schauß und PD Dr. David Sabel

Noch ein Beispiel

Berechne Typ von `(map head)`

`map`::	$(a \to b)$	$\to \ [a] \to [b]$
`head`::	$[a] \to a$	
Gesuchter Typ:		$\gamma([a] \to [b])$

Regelanwendung benötigt Lösung γ von $(a \to b) \doteq ([a'] \to a')$:

G	E

Noch ein Beispiel

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

Berechne Typ von `(map head)`

`map::`	$(a \to b)$	$\to ([a] \to [b])$
`head::`	$[a] \to a$	
Gesuchter Typ:		$\gamma([a] \to [b])$

Regelanwendung benötigt Lösung γ von $(a \to b) \doteq ([a'] \to a')$:

G	E
$\emptyset;$	$\{(a \to b) \doteq ([a'] \to a')\}$

Noch ein Beispiel

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

Berechne Typ von `(map head)`

`map::`	$(a \to b)$	$\to ([a] \to [b])$
`head::`	$[a] \to a$	
Gesuchter Typ:		$\gamma([a] \to [b])$

Regelanwendung benötigt Lösung γ von $(a \to b) \doteq ([a'] \to a')$:

G	E
$\emptyset$;	$\{(a \to b) \doteq ([a'] \to a')\}$

Noch ein Beispiel

Berechne Typ von `(map head)`

`map::`	$(a \to b)$	$\to ([a] \to [b])$
`head::`	$[a] \to a$	
Gesuchter Typ:		$\gamma([a] \to [b])$

Regelanwendung benötigt Lösung γ von $(a \to b) \doteq ([a'] \to a')$:

G	E
$\emptyset;$	$\{(a \to b) \doteq ([a'] \to a')\}$
$\emptyset;$	$\{a \doteq [a'], b \doteq a'\}$

Noch ein Beispiel

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

Berechne Typ von `(map head)`

`map`::	$(a \to b)$	$\to ([a] \to [b])$
`head`::	$[a] \to a$	
Gesuchter Typ:		$\gamma([a] \to [b])$

Regelanwendung benötigt Lösung γ von $(a \to b) \doteq ([a'] \to a')$:

G	E
$\emptyset;$	$\{(a \to b) \doteq ([a'] \to a')\}$
$\emptyset;$	$\{a \doteq [a'], b \doteq a'\}$

Noch ein Beispiel

Berechne Typ von `(map head)`

`map::`	$(a \to b)$	$\to ([a] \to [b])$
`head::`	$[a] \to a$	
Gesuchter Typ:		$\gamma([a] \to [b])$

Regelanwendung benötigt Lösung γ von $(a \to b) \doteq ([a'] \to a')$:

G	E
$\emptyset;$	$\{(a \to b) \doteq ([a'] \to a')\}$
$\emptyset;$	$\{a \doteq [a'], b \doteq a'\}$
$\{a \mapsto [a']\};$	$\{b \doteq a'\}$

Noch ein Beispiel

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

Berechne Typ von `(map head)`

`map::`	$(a \to b)$	$\to ([a] \to [b])$
`head::`	$[a] \to a$	
Gesuchter Typ:		$\gamma([a] \to [b])$

Regelanwendung benötigt Lösung γ von $(a \to b) \doteq ([a'] \to a')$:

G	E
$\emptyset;$	$\{(a \to b) \doteq ([a'] \to a')\}$
$\emptyset;$	$\{a \doteq [a'], b \doteq a'\}$
$\{a \mapsto [a']\};$	$\{b \doteq a'\}$

Noch ein Beispiel

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

Berechne Typ von `(map head)`

`map::`	$(a \to b)$	$\to ([a] \to [b])$
`head::`	$[a] \to a$	
Gesuchter Typ:		$\gamma([a] \to [b])$

Regelanwendung benötigt Lösung γ von $(a \to b) \doteq ([a'] \to a')$:

G	E
$\emptyset;$	$\{(a \to b) \doteq ([a'] \to a')\}$
$\emptyset;$	$\{a \doteq [a'], b \doteq a'\}$
$\{a \mapsto [a']\};$	$\{b \doteq a'\}$
$\{a \mapsto [a'], b \mapsto a'\};$	$\emptyset$

Noch ein Beispiel

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

Berechne Typ von `(map head)`

`map::`	$(a \to b)$	$\to ([a] \to [b])$
`head::`	$[a] \to a$	
Gesuchter Typ:		$\gamma([a] \to [b])$

Regelanwendung benötigt Lösung γ von $(a \to b) \doteq ([a'] \to a')$:

G	E
$\emptyset$;	$\{(a \to b) \doteq ([a'] \to a')\}$
$\emptyset$;	$\{a \doteq [a'], b \doteq a'\}$
$\{a \mapsto [a']\}$;	$\{b \doteq a'\}$
$\{a \mapsto [a'], b \mapsto a'\}$;	$\emptyset$
$\{a \mapsto [a'], b \mapsto a'\}$;	

Noch ein Beispiel

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

Berechne Typ von `(map head)`

`map`::	$(a \to b)$	$\to ([a] \to [b])$
`head`::	$[a] \to a$	
Gesuchter Typ:		$\gamma([a] \to [b])$

Regelanwendung benötigt Lösung γ von $(a \to b) \doteq ([a'] \to a')$:

G	E
$\emptyset$;	$\{(a \to b) \doteq ([a'] \to a')\}$
$\emptyset$;	$\{a \doteq [a'], b \doteq a'\}$
$\{a \mapsto [a']\}$;	$\{b \doteq a'\}$
$\{a \mapsto [a'], b \mapsto a'\}$;	$\emptyset$
$\{a \mapsto [a'], b \mapsto a'\}$;	

Einsetzen der Lösung $\gamma = \{a \mapsto [a'], b \mapsto a'\}$
in $[a] \to [b]$ ergibt:
`(map head)` :: $([[a']] \to [a'])$.

Beispiel zu Typberechnung

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

Typ von `map length`

$$\frac{\texttt{map} :: (a \to b) \to ([a] \to [b]), \quad \texttt{length} :: [a'] \to \texttt{Int}}{(\texttt{map length}) :: ? = \gamma([a] \to [b])}$$

Unifiziere $(a \to b) \doteq ([a'] \to \texttt{Int})$

G	E
$\emptyset$;	$\{(a \to b) \doteq ([a'] \to \texttt{Int})\}$

Beispiel zu Typberechnung

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

Typ von `map length`

$$\frac{\texttt{map} :: (a \to b) \to ([a] \to [b]), \quad \texttt{length} :: [a'] \to \texttt{Int}}{(\texttt{map length}) :: ? = \gamma([a] \to [b])}$$

Unifiziere $(a \to b) \doteq ([a'] \to \texttt{Int})$

G	E
$\emptyset;$	$\{(a \to b) \doteq ([a'] \to \texttt{Int})\}$
$\emptyset;$	$\{a \doteq [a'], b \doteq \texttt{Int}\}$

Beispiel zu Typberechnung

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

Typ von `map length`

$$\frac{\texttt{map} :: (a \to b) \to ([a] \to [b]), \quad \texttt{length} :: [a'] \to \texttt{Int}}{(\texttt{map length}) :: ? = \gamma([a] \to [b])}$$

Unifiziere $(a \to b) \doteq ([a'] \to \texttt{Int})$

G	E
$\emptyset;$	$\{(a \to b) \doteq ([a'] \to \texttt{Int})\}$
$\emptyset;$	$\{a \doteq [a'], b \doteq \texttt{Int}\}$

Beispiel zu Typberechnung

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

Typ von `map length`

$$\frac{\texttt{map} :: (a \to b) \to ([a] \to [b]), \quad \texttt{length} :: [a'] \to \texttt{Int}}{(\texttt{map length}) :: ? = \gamma([a] \to [b])}$$

Unifiziere $(a \to b) \doteq ([a'] \to \texttt{Int})$

G	E
$\emptyset$;	$\{(a \to b) \doteq ([a'] \to \texttt{Int})\}$
$\emptyset$;	$\{a \doteq [a'], b \doteq \texttt{Int}\}$
$\{a \mapsto [a']\}$;	$\{b \doteq \texttt{Int}\}$

Beispiel zu Typberechnung

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

Typ von `map length`

$$\frac{\texttt{map} :: (a \to b) \to ([a] \to [b]), \quad \texttt{length} :: [a'] \to \texttt{Int}}{(\texttt{map length}) :: ? = \gamma([a] \to [b])}$$

Unifiziere $(a \to b) \doteq ([a'] \to \texttt{Int})$

G	E
$\emptyset$;	$\{(a \to b) \doteq ([a'] \to \texttt{Int})\}$
$\emptyset$;	$\{a \doteq [a'], b \doteq \texttt{Int}\}$
$\{a \mapsto [a']\}$;	$\{b \doteq \texttt{Int}\}$
$\{a \mapsto [a'], b \mapsto \texttt{Int}\}$;	$\emptyset$

Beispiel zu Typberechnung

Typ von `map length`

$$\frac{\texttt{map} :: (a \to b) \to ([a] \to [b]), \quad \texttt{length} :: [a'] \to \texttt{Int}}{(\texttt{map length}) :: ? = \gamma([a] \to [b])}$$

Unifiziere $(a \to b) \doteq ([a'] \to \texttt{Int})$

G	E
$\emptyset;$	$\{(a \to b) \doteq ([a'] \to \texttt{Int})\}$
$\emptyset;$	$\{a \doteq [a'], b \doteq \texttt{Int}\}$
$\{a \mapsto [a']\};$	$\{b \doteq \texttt{Int}\}$
$\{a \mapsto [a'], b \mapsto \texttt{Int}\};$	$\emptyset$

Somit: $(\texttt{map length}) :: \gamma([a] \to [b]) = [[a']] \to [\texttt{Int}]$

Beispiel `(foldr (:) [])` :: ??

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

```
foldr :: (a -> b -> b) -> b -> [a] -> b
(:)   :: a -> [a] -> [a]
([])  :: [a]
```

G	E
$\emptyset$	

Beispiel `(foldr (:) [])` :: ??

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

```
foldr :: (a -> b -> b) -> b -> [a] -> b
(:)   :: a -> [a] -> [a]     umbenannt: c -> [c] -> [c]
([])  :: [a]                 umbenannt: [d]
```

G	E
$\emptyset$	$\{a \to b \to b \doteq c \to [c] \to [c], b \doteq [d]\}$

Beispiel `(foldr (:) [])` :: ??

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

```
foldr :: (a -> b -> b) -> b -> [a] -> b
(:)   :: a -> [a] -> [a]     umbenannt: c -> [c] -> [c]
([])  :: [a]                 umbenannt: [d]
```

G	E
$\emptyset$	$\{a \to b \to b \doteq c \to [c] \to [c], b \doteq [d]\}$

Beispiel `(foldr (:) [])` :: ??

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

```
foldr :: (a -> b -> b) -> b -> [a] -> b
(:)   :: a -> [a] -> [a]     umbenannt: c -> [c] -> [c]
([])  :: [a]                 umbenannt: [d]
```

G	E
$\emptyset$	$\{a \to b \to b \doteq c \to [c] \to [c], b \doteq [d]\}$
$\{b \mapsto [d]\}$	$\{a \to [d] \to [d] \doteq c \to [c] \to [c]\}$

Beispiel `(foldr (:) [])` :: ??

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

```
foldr :: (a -> b -> b) -> b -> [a] -> b
(:)   :: a -> [a] -> [a]     umbenannt: c -> [c] -> [c]
([])  :: [a]                 umbenannt: [d]
```

G	E
$\emptyset$	$\{a \to b \to b \doteq c \to [c] \to [c], b \doteq [d]\}$
$\{b \mapsto [d]\}$	$\{a \to [d] \to [d] \doteq c \to [c] \to [c]\}$

Beispiel `(foldr (:) [])` :: ??

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

```
foldr :: (a -> b -> b) -> b -> [a] -> b
(:)   :: a -> [a] -> [a]      umbenannt: c -> [c] -> [c]
([])  :: [a]                 umbenannt: [d]
```

G	E
$\emptyset$	$\{a \to b \to b \doteq c \to [c] \to [c], b \doteq [d]\}$
$\{b \mapsto [d]\}$	$\{a \to [d] \to [d] \doteq c \to [c] \to [c]\}$
$\{b \mapsto [d]\}$	$\{a \doteq c, [d] \doteq [c], [d] \doteq [c]\}$

Beispiel `(foldr (:) [])` :: ??

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

```
foldr :: (a -> b -> b) -> b -> [a] -> b
(:)   :: a -> [a] -> [a]      umbenannt: c -> [c] -> [c]
([])  :: [a]                 umbenannt: [d]
```

G	E
$\emptyset$	$\{a \to b \to b \doteq c \to [c] \to [c], b \doteq [d]\}$
$\{b \mapsto [d]\}$	$\{a \to [d] \to [d] \doteq c \to [c] \to [c]\}$
$\{b \mapsto [d]\}$	$\{a \doteq c, [d] \doteq [c], [d] \doteq [c]\}$

Beispiel `(foldr (:) [])` :: ??

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

```
foldr :: (a -> b -> b) -> b -> [a] -> b
(:)   :: a -> [a] -> [a]      umbenannt: c -> [c] -> [c]
([])  :: [a]                 umbenannt: [d]
```

G	E
$\emptyset$	$\{a \to b \to b \doteq c \to [c] \to [c], b \doteq [d]\}$
$\{b \mapsto [d]\}$	$\{a \to [d] \to [d] \doteq c \to [c] \to [c]\}$
$\{b \mapsto [d]\}$	$\{a \doteq c, [d] \doteq [c], [d] \doteq [c]\}$
$\{b \mapsto [d], a \mapsto c\}$	$\{[d] \doteq [c], [d] \doteq [c]\}$

Beispiel `(foldr (:) [])` :: ??

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

```
foldr :: (a -> b -> b) -> b -> [a] -> b
(:)   :: a -> [a] -> [a]      umbenannt: c -> [c] -> [c]
([])  :: [a]                 umbenannt: [d]
```

G	E
$\emptyset$	$\{a \to b \to b \doteq c \to [c] \to [c], b \doteq [d]\}$
$\{b \mapsto [d]\}$	$\{a \to [d] \to [d] \doteq c \to [c] \to [c]\}$
$\{b \mapsto [d]\}$	$\{a \doteq c, [d] \doteq [c], [d] \doteq [c]\}$
$\{b \mapsto [d], a \mapsto c\}$	$\{[d] \doteq [c], [d] \doteq [c]\}$

Beispiel `(foldr (:) [])` :: ??

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

```
foldr :: (a -> b -> b) -> b -> [a] -> b
(:)   :: a -> [a] -> [a]      umbenannt: c -> [c] -> [c]
([])  :: [a]                  umbenannt: [d]
```

G	E
$\emptyset$	$\{a \to b \to b \doteq c \to [c] \to [c], b \doteq [d]\}$
$\{b \mapsto [d]\}$	$\{a \to [d] \to [d] \doteq c \to [c] \to [c]\}$
$\{b \mapsto [d]\}$	$\{a \doteq c, [d] \doteq [c], [d] \doteq [c]\}$
$\{b \mapsto [d], a \mapsto c\}$	$\{[d] \doteq [c], [d] \doteq [c]\}$
$\{b \mapsto [d], a \mapsto c\}$	$\{d \doteq c, [d] \doteq [c]\}$

Beispiel `(foldr (:) [])` :: ??

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

```
foldr :: (a -> b -> b) -> b -> [a] -> b
(:)   :: a -> [a] -> [a]      umbenannt: c -> [c] -> [c]
([])  :: [a]                  umbenannt: [d]
```

G	E
$\emptyset$	$\{a \to b \to b \doteq c \to [c] \to [c], b \doteq [d]\}$
$\{b \mapsto [d]\}$	$\{a \to [d] \to [d] \doteq c \to [c] \to [c]\}$
$\{b \mapsto [d]\}$	$\{a \doteq c, [d] \doteq [c], [d] \doteq [c]\}$
$\{b \mapsto [d], a \mapsto c\}$	$\{[d] \doteq [c], [d] \doteq [c]\}$
$\{b \mapsto [d], a \mapsto c\}$	$\{d \doteq c, [d] \doteq [c]\}$

Beispiel `(foldr (:) [])` :: ??

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

```
foldr :: (a -> b -> b) -> b -> [a] -> b
(:)   :: a -> [a] -> [a]     umbenannt: c -> [c] -> [c]
([])  :: [a]                 umbenannt: [d]
```

G	E
$\emptyset$	$\{a \to b \to b \doteq c \to [c] \to [c], b \doteq [d]\}$
$\{b \mapsto [d]\}$	$\{a \to [d] \to [d] \doteq c \to [c] \to [c]\}$
$\{b \mapsto [d]\}$	$\{a \doteq c, [d] \doteq [c], [d] \doteq [c]\}$
$\{b \mapsto [d], a \mapsto c\}$	$\{[d] \doteq [c], [d] \doteq [c]\}$
$\{b \mapsto [d], a \mapsto c\}$	$\{d \doteq c, [d] \doteq [c]\}$
$\{b \mapsto [c], a \mapsto c, d \mapsto c\}$	$\{[c] \doteq [c]\}$

Beispiel `(foldr (:) [])` :: ??

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

```
foldr :: (a -> b -> b) -> b -> [a] -> b
(:)   :: a -> [a] -> [a]      umbenannt: c -> [c] -> [c]
([])  :: [a]                  umbenannt: [d]
```

G	E
$\emptyset$	$\{a \to b \to b \doteq c \to [c] \to [c], b \doteq [d]\}$
$\{b \mapsto [d]\}$	$\{a \to [d] \to [d] \doteq c \to [c] \to [c]\}$
$\{b \mapsto [d]\}$	$\{a \doteq c, [d] \doteq [c], [d] \doteq [c]\}$
$\{b \mapsto [d], a \mapsto c\}$	$\{[d] \doteq [c], [d] \doteq [c]\}$
$\{b \mapsto [d], a \mapsto c\}$	$\{d \doteq c, [d] \doteq [c]\}$
$\{b \mapsto [c], a \mapsto c, d \mapsto c\}$	$\{[c] \doteq [c]\}$

Beispiel `(foldr (:) [])` :: ??

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

```
foldr :: (a -> b -> b) -> b -> [a] -> b
(:)   :: a -> [a] -> [a]      umbenannt: c -> [c] -> [c]
([])  :: [a]                 umbenannt: [d]
```

G	E
$\emptyset$	$\{a \to b \to b \doteq c \to [c] \to [c], b \doteq [d]\}$
$\{b \mapsto [d]\}$	$\{a \to [d] \to [d] \doteq c \to [c] \to [c]\}$
$\{b \mapsto [d]\}$	$\{a \doteq c, [d] \doteq [c], [d] \doteq [c]\}$
$\{b \mapsto [d], a \mapsto c\}$	$\{[d] \doteq [c], [d] \doteq [c]\}$
$\{b \mapsto [d], a \mapsto c\}$	$\{d \doteq c, [d] \doteq [c]\}$
$\{b \mapsto [c], a \mapsto c, d \mapsto c\}$	$\{[c] \doteq [c]\}$
$\{b \mapsto [c], a \mapsto c, d \mapsto c\}$	$\{c \doteq c\}$

Beispiel `(foldr (:) [])` :: ??

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

```
foldr :: (a -> b -> b) -> b -> [a] -> b
(:)   :: a -> [a] -> [a]      umbenannt: c -> [c] -> [c]
([])  :: [a]                 umbenannt: [d]
```

G	E
$\emptyset$	$\{a \to b \to b \doteq c \to [c] \to [c], b \doteq [d]\}$
$\{b \mapsto [d]\}$	$\{a \to [d] \to [d] \doteq c \to [c] \to [c]\}$
$\{b \mapsto [d]\}$	$\{a \doteq c, [d] \doteq [c], [d] \doteq [c]\}$
$\{b \mapsto [d], a \mapsto c\}$	$\{[d] \doteq [c], [d] \doteq [c]\}$
$\{b \mapsto [d], a \mapsto c\}$	$\{d \doteq c, [d] \doteq [c]\}$
$\{b \mapsto [c], a \mapsto c, d \mapsto c\}$	$\{[c] \doteq [c]\}$
$\{b \mapsto [c], a \mapsto c, d \mapsto c\}$	$\{c \doteq c\}$
$\{b \mapsto [c], a \mapsto c, d \mapsto c\}$	$\{\}$

Beispiel `(foldr (:) [])` :: ??

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

```
foldr :: (a -> b -> b) -> b -> [a] -> b
(:)   :: a -> [a] -> [a]      umbenannt: c -> [c] -> [c]
([])  :: [a]                  umbenannt: [d]
```

G	E
$\emptyset$	$\{a \to b \to b \doteq c \to [c] \to [c], b \doteq [d]\}$
$\{b \mapsto [d]\}$	$\{a \to [d] \to [d] \doteq c \to [c] \to [c]\}$
$\{b \mapsto [d]\}$	$\{a \doteq c, [d] \doteq [c], [d] \doteq [c]\}$
$\{b \mapsto [d], a \mapsto c\}$	$\{[d] \doteq [c], [d] \doteq [c]\}$
$\{b \mapsto [d], a \mapsto c\}$	$\{d \doteq c, [d] \doteq [c]\}$
$\{b \mapsto [c], a \mapsto c, d \mapsto c\}$	$\{[c] \doteq [c]\}$
$\{b \mapsto [c], a \mapsto c, d \mapsto c\}$	$\{c \doteq c\}$
$\{b \mapsto [c], a \mapsto c, d \mapsto c\}$	$\{\}$

`(foldr (:) [])`$::[c] \to [c] \qquad = \gamma([a] \to b)$

Beispiel. Linksfaltung: `(foldl (:) [])`?

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

```
foldl :: (a -> b -> a) -> a -> [b] -> a
(:)   :: a -> [a] -> [a]
([])  :: [a]
```

G	E
$\emptyset$	

Beispiel. Linksfaltung: `(foldl (:) [])`?

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

```
foldl :: (a -> b -> a) -> a -> [b] -> a
(:)   :: a -> [a] -> [a]     umbenannt: c -> [c] -> [c]
([])  :: [a]                 umbenannt: [d]
```

G	E
$\emptyset$	$\{a \to b \to a \doteq c \to [c] \to [c], a \doteq [d]\}$

Beispiel. Linksfaltung: `(foldl (:) [])`?

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

```
foldl :: (a -> b -> a) -> a -> [b] -> a
(:)   :: a -> [a] -> [a]      umbenannt: c -> [c] -> [c]
([])  :: [a]                 umbenannt: [d]
```

G	E
$\emptyset$	$\{a \to b \to a \doteq c \to [c] \to [c], a \doteq [d]\}$

Beispiel. Linksfaltung: `(foldl (:) [])`?

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

```
foldl :: (a -> b -> a) -> a -> [b] -> a
(:)   :: a -> [a] -> [a]      umbenannt: c -> [c] -> [c]
([])  :: [a]                 umbenannt: [d]
```

G	E
$\emptyset$	$\{a \to b \to a \doteq c \to [c] \to [c], a \doteq [d]\}$
$\{a \mapsto [d]\}$	$\{[d] \to b \to [d] \doteq c \to [c] \to [c]\}$

Beispiel. Linksfaltung: `(foldl (:) [])`?

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

```
foldl :: (a -> b -> a) -> a -> [b] -> a
(:)   :: a -> [a] -> [a]      umbenannt: c -> [c] -> [c]
([])  :: [a]                 umbenannt: [d]
```

G	E
$\emptyset$	$\{a \to b \to a \doteq c \to [c] \to [c], a \doteq [d]\}$
$\{a \mapsto [d]\}$	$\{[d] \to b \to [d] \doteq c \to [c] \to [c]\}$

Beispiel. Linksfaltung: `(foldl (:) [])`?

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

```
foldl :: (a -> b -> a) -> a -> [b] -> a
(:)   :: a -> [a] -> [a]      umbenannt: c -> [c] -> [c]
([])  :: [a]                 umbenannt: [d]
```

G	E
$\emptyset$	$\{a \to b \to a \doteq c \to [c] \to [c], a \doteq [d]\}$
$\{a \mapsto [d]\}$	$\{[d] \to b \to [d] \doteq c \to [c] \to [c]\}$
$\{a \mapsto [d]\}$	$\{[d] \doteq c, b \doteq [c], [d] \doteq [c]\}$

Beispiel. Linksfaltung: `(foldl (:) [])`?

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

```
foldl :: (a -> b -> a) -> a -> [b] -> a
(:)   :: a -> [a] -> [a]      umbenannt: c -> [c] -> [c]
([])  :: [a]                 umbenannt: [d]
```

G	E
$\emptyset$	$\{a \to b \to a \doteq c \to [c] \to [c], a \doteq [d]\}$
$\{a \mapsto [d]\}$	$\{[d] \to b \to [d] \doteq c \to [c] \to [c]\}$
$\{a \mapsto [d]\}$	$\{[d] \doteq c, b \doteq [c], [d] \doteq [c]\}$

Beispiel. Linksfaltung: `(foldl (:) [])`?

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

```
foldl :: (a -> b -> a) -> a -> [b] -> a
(:)   :: a -> [a] -> [a]      umbenannt: c -> [c] -> [c]
([])  :: [a]                 umbenannt: [d]
```

G	E
$\emptyset$	$\{a \to b \to a \doteq c \to [c] \to [c], a \doteq [d]\}$
$\{a \mapsto [d]\}$	$\{[d] \to b \to [d] \doteq c \to [c] \to [c]\}$
$\{a \mapsto [d]\}$	$\{[d] \doteq c, b \doteq [c], [d] \doteq [c]\}$
$\{a \mapsto [d], b \mapsto [c]\}$	$\{[d] \doteq c, [d] \doteq [c]\}$

Beispiel. Linksfaltung: `(foldl (:) [])`?

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

```
foldl :: (a -> b -> a) -> a -> [b] -> a
(:)   :: a -> [a] -> [a]      umbenannt: c -> [c] -> [c]
([])  :: [a]                 umbenannt: [d]
```

G	E
$\emptyset$	$\{a \to b \to a \doteq c \to [c] \to [c], a \doteq [d]\}$
$\{a \mapsto [d]\}$	$\{[d] \to b \to [d] \doteq c \to [c] \to [c]\}$
$\{a \mapsto [d]\}$	$\{[d] \doteq c, b \doteq [c], [d] \doteq [c]\}$
$\{a \mapsto [d], b \mapsto [c]\}$	$\{[d] \doteq c, [d] \doteq [c]\}$

Beispiel. Linksfaltung: `(foldl (:) [])`?

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

```
foldl :: (a -> b -> a) -> a -> [b] -> a
(:)   :: a -> [a] -> [a]      umbenannt: c -> [c] -> [c]
([])  :: [a]                 umbenannt: [d]
```

G	E
$\emptyset$	$\{a \to b \to a \doteq c \to [c] \to [c], a \doteq [d]\}$
$\{a \mapsto [d]\}$	$\{[d] \to b \to [d] \doteq c \to [c] \to [c]\}$
$\{a \mapsto [d]\}$	$\{[d] \doteq c, b \doteq [c], [d] \doteq [c]\}$
$\{a \mapsto [d], b \mapsto [c]\}$	$\{[d] \doteq c, [d] \doteq [c]\}$
$\{a \mapsto [d], b \mapsto [c]\}$	$\{c \doteq [d], [d] \doteq [c]\}$

Beispiel. Linksfaltung: `(foldl (:) [])`?

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

```
foldl :: (a -> b -> a) -> a -> [b] -> a
(:)   :: a -> [a] -> [a]      umbenannt: c -> [c] -> [c]
([])  :: [a]                 umbenannt: [d]
```

G	E
$\emptyset$	$\{a \to b \to a \doteq c \to [c] \to [c], a \doteq [d]\}$
$\{a \mapsto [d]\}$	$\{[d] \to b \to [d] \doteq c \to [c] \to [c]\}$
$\{a \mapsto [d]\}$	$\{[d] \doteq c, b \doteq [c], [d] \doteq [c]\}$
$\{a \mapsto [d], b \mapsto [c]\}$	$\{[d] \doteq c, [d] \doteq [c]\}$
$\{a \mapsto [d], b \mapsto [c]\}$	$\{c \doteq [d], [d] \doteq [c]\}$

Beispiel. Linksfaltung: `(foldl (:) [])`?

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

```
foldl :: (a -> b -> a) -> a -> [b] -> a
(:)   :: a -> [a] -> [a]      umbenannt: c -> [c] -> [c]
([])  :: [a]                 umbenannt: [d]
```

G	E
$\emptyset$	$\{a \to b \to a \doteq c \to [c] \to [c], a \doteq [d]\}$
$\{a \mapsto [d]\}$	$\{[d] \to b \to [d] \doteq c \to [c] \to [c]\}$
$\{a \mapsto [d]\}$	$\{[d] \doteq c, b \doteq [c], [d] \doteq [c]\}$
$\{a \mapsto [d], b \mapsto [c]\}$	$\{[d] \doteq c, [d] \doteq [c]\}$
$\{a \mapsto [d], b \mapsto [c]\}$	$\{c \doteq [d], [d] \doteq [c]\}$
$\{a \mapsto [d], b \mapsto [[d]], c \mapsto [d]\}$	$\{[d] \doteq [[d]]\}$

Beispiel. Linksfaltung: `(foldl (:) [])`?

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

```
foldl :: (a -> b -> a) -> a -> [b] -> a
(:)   :: a -> [a] -> [a]      umbenannt: c -> [c] -> [c]
([])  :: [a]                 umbenannt: [d]
```

G	E
$\emptyset$	$\{a \to b \to a \doteq c \to [c] \to [c], a \doteq [d]\}$
$\{a \mapsto [d]\}$	$\{[d] \to b \to [d] \doteq c \to [c] \to [c]\}$
$\{a \mapsto [d]\}$	$\{[d] \doteq c, b \doteq [c], [d] \doteq [c]\}$
$\{a \mapsto [d], b \mapsto [c]\}$	$\{[d] \doteq c, [d] \doteq [c]\}$
$\{a \mapsto [d], b \mapsto [c]\}$	$\{c \doteq [d], [d] \doteq [c]\}$
$\{a \mapsto [d], b \mapsto [[d]], c \mapsto [d]\}$	$\{[d] \doteq [[d]]\}$

Beispiel. Linksfaltung: `(foldl (:) [])`?

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

```
foldl :: (a -> b -> a) -> a -> [b] -> a
(:)   :: a -> [a] -> [a]      umbenannt: c -> [c] -> [c]
([])  :: [a]                 umbenannt: [d]
```

G	E
$\emptyset$	$\{a \to b \to a \doteq c \to [c] \to [c], a \doteq [d]\}$
$\{a \mapsto [d]\}$	$\{[d] \to b \to [d] \doteq c \to [c] \to [c]\}$
$\{a \mapsto [d]\}$	$\{[d] \doteq c, b \doteq [c], [d] \doteq [c]\}$
$\{a \mapsto [d], b \mapsto [c]\}$	$\{[d] \doteq c, [d] \doteq [c]\}$
$\{a \mapsto [d], b \mapsto [c]\}$	$\{c \doteq [d], [d] \doteq [c]\}$
$\{a \mapsto [d], b \mapsto [[d]], c \mapsto [d]\}$	$\{[d] \doteq [[d]]\}$
$\{a \mapsto [d], b \mapsto [[d]], c \mapsto [d]\}$	$\{d \doteq [d]\}$

Beispiel. Linksfaltung: `(foldl (:) [])`?

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

```
foldl :: (a -> b -> a) -> a -> [b] -> a
(:)   :: a -> [a] -> [a]      umbenannt: c -> [c] -> [c]
([])  :: [a]                 umbenannt: [d]
```

G	E
$\emptyset$	$\{a \to b \to a \doteq c \to [c] \to [c], a \doteq [d]\}$
$\{a \mapsto [d]\}$	$\{[d] \to b \to [d] \doteq c \to [c] \to [c]\}$
$\{a \mapsto [d]\}$	$\{[d] \doteq c, b \doteq [c], [d] \doteq [c]\}$
$\{a \mapsto [d], b \mapsto [c]\}$	$\{[d] \doteq c, [d] \doteq [c]\}$
$\{a \mapsto [d], b \mapsto [c]\}$	$\{c \doteq [d], [d] \doteq [c]\}$
$\{a \mapsto [d], b \mapsto [[d]], c \mapsto [d]\}$	$\{[d] \doteq [[d]]\}$
$\{a \mapsto [d], b \mapsto [[d]], c \mapsto [d]\}$	$\{d \doteq [d]\}$

nicht lösbar, da d in $[d]$ echt vorkommt

`(foldl (:) [])` ist nicht typisierbar!

Listen müssen Elemente gleichen Typs haben

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

Positivbeispiel: Berechne Typ der Liste $[1]$:

$$\frac{1 :: \texttt{Int} \qquad (:) :: a \to [a] \to [a] \qquad [] :: [b]}{1 : [] ::?}$$

Anwendungsregel ergibt Gleichungen: $\{a \doteq \texttt{Int}, [a] \doteq [b]\}$

Lösung: $\gamma = \{a \mapsto \texttt{Int}, b \mapsto \texttt{Int}\}$

Anwenden auf $[a]$:

Typ von $(1 : [])$ ist $[\texttt{Int}]$

Beispiel zu Typfehler

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

Negativbeispiel: $[1,\ \texttt{'a'}]$ hat keinen Typ:

- $1 : (\texttt{'a'} : []) = [1,\ \texttt{'a'}]$
- $1 :: \texttt{Integer}$, $\texttt{'a'} :: \texttt{Char}$, $[] :: [b]$, $(:)$ hat Typ $a \to [a] \to [a]$

(Typen der Konstanten.)

ergibt: $(1 :) :: [\texttt{Integer}] \to [\texttt{Integer}]$ und
$\texttt{('a':[])} :: [\texttt{Char}]$.

Kein Typ als Resultat, denn:

$[\texttt{Integer}] \doteq [\texttt{Char}]$ ist nicht lösbar.

Typisierung und Reduktion

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

Beachte: Nach Reduktionen kann ein Ausdruck
mehr Typen (bzw. einen allgemeineren Typ) haben
als vor der Reduktion

Typisierung und Reduktion

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

Beachte: Nach Reduktionen kann ein Ausdruck mehr Typen (bzw. einen allgemeineren Typ) haben als vor der Reduktion

Beispiel:

```
if 1 > 0 then [] else [1]      :: [Integer]
```

arithmetische-Reduktion:

```
⟶ if True then [] else [1]    :: [Integer]
```

Case-Reduktion:

```
⟶ []                          :: [a]
```