

Algoritmické typování

IB015 Neimperativní programování

Kolektiv cvičících IB015

Fakulta informatiky, Masarykova univerzita

podzim 2018

Intuitivní typování

výraz posoudíme z hlediska toho, co dělá, a na základě toho určíme typ

vhodné pro jednoduché výrazy

Algoritmické typování

exaktní určení typu typovacím algoritmem

zdlouhavější, ale funguje pro všechny výrazy

Algoritmické typování: příklad I

Jaký typ má výraz `map head . snd`?

Algoritmické typování: příklad I

Jaký typ má výraz `map head . snd`?

Jaký typ má `map head`?

Jaký typ má `snd`?

Jaký typ má `(.)`?

Algoritmické typování: příklad I

Jaký typ má výraz `map head . snd`?

Jaký typ má `map head`?

Jaký typ má `head`?

Jaký typ má `map`?

Jaký typ má `snd`?

Jaký typ má `(.)`?

Algoritmické typování: příklad I

Jaký typ má výraz `map head . snd`?

Jaký typ má `map head`?

Jaký typ má `head`? `[a] -> a`

Jaký typ má `map`?

Jaký typ má `snd`?

Jaký typ má `(.)`?

Algoritmické typování: příklad I

Jaký typ má výraz `map head . snd`?

Jaký typ má `map head`?

Jaký typ má `head`? `[a] -> a`

Jaký typ má `map`? `(b -> c) -> [b] -> [c]`

Jaký typ má `snd`?

Jaký typ má `(.)`?

Algoritmické typování: příklad I

Jaký typ má výraz `map head . snd`?

Jaký typ má `map head`?

Jaký typ má `head`? `[a] -> a`

Jaký typ má `map`? `(b -> c) -> [b] -> [c]`

`map` $::$ `(b -> c) -> [b] -> [c]`

`head` $::$ `([a] -> a)`

Jaký typ má `snd`?

Jaký typ má `(.)`?

Algoritmické typování: příklad I

Jaký typ má výraz `map head . snd`?

Jaký typ má `map head`?

Jaký typ má `head`? `[a] -> a`

Jaký typ má `map`? `(b -> c) -> [b] -> [c]`

Rovnosti:

`map` `::` `(b -> c) -> [b] -> [c]`

`head` `::` `([a] -> a)`

Jaký typ má `snd`?

Jaký typ má `(.)`?

Algoritmické typování: příklad I

Jaký typ má výraz `map head . snd`?

Jaký typ má `map head`?

Jaký typ má `head`? `[a] -> a`

Jaký typ má `map`? `(b -> c) -> [b] -> [c]`

Rovnosti: `b = [a]`

`map` `::` `(b -> c) -> [b] -> [c]`

`head` `::` `([a] -> a)`

Jaký typ má `snd`?

Jaký typ má `(.)`?

Algoritmické typování: příklad I

Jaký typ má výraz `map head . snd`?

Jaký typ má `map head`?

Jaký typ má `head`? `[a] -> a`

Jaký typ má `map`? `(b -> c) -> [b] -> [c]`

Rovnosti: `b = [a]` , `c = a`

`map` `::` `(b -> c) -> [b] -> [c]`

`head` `::` `([a] -> a)`

Jaký typ má `snd`?

Jaký typ má `(.)`?

Algoritmické typování: příklad I

Jaký typ má výraz `map head . snd`?

Jaký typ má `map head`?

Jaký typ má `head`? `[a] -> a`

Jaký typ má `map`? `(b -> c) -> [b] -> [c]`

Rovnosti: `b = [a]` , `c = a`

<code>map</code>	<code>::</code>	<code>(b -> c)</code>	<code>-></code>	<code>[b]</code>	<code>-></code>	<code>[c]</code>
<code>map</code>	<code>::</code>	<code>([a] -> a)</code>	<code>-></code>	<code>[[a]]</code>	<code>-></code>	<code>[a]</code>
<code>head</code>	<code>::</code>	<code>([a] -> a)</code>				

Jaký typ má `snd`?

Jaký typ má `(.)`?

Algoritmické typování: příklad I

Jaký typ má výraz `map head . snd`?

Jaký typ má `map head`?

Jaký typ má `head`? `[a] -> a`

Jaký typ má `map`? `(b -> c) -> [b] -> [c]`

Rovnosti: `b = [a]` , `c = a`

<code>map</code>	<code>::</code>	<code>(b -> c)</code>	<code>-></code>	<code>[b]</code>	<code>-></code>	<code>[c]</code>
<code>map</code>	<code>::</code>	<code>([a] -> a)</code>	<code>-></code>	<code>[[a]]</code>	<code>-></code>	<code>[a]</code>
<code>head</code>	<code>::</code>	<code>([a] -> a)</code>				
<hr/>						
<code>map head</code>	<code>::</code>			<code>[[a]]</code>	<code>-></code>	<code>[a]</code>

Jaký typ má `snd`?

Jaký typ má `(.)`?

Algoritmické typování: příklad I

Jaký typ má výraz `map head . snd`?

Jaký typ má `map head`? `[[a]] -> a`

Jaký typ má `snd`?

Jaký typ má `(.)`?

Algoritmické typování: příklad I

Jaký typ má výraz `map head . snd`?

Jaký typ má `map head`? `[[a]] -> a`

Jaký typ má `snd`? `(b, c) -> c`

Jaký typ má `(.)`?

Algoritmické typování: příklad I

Jaký typ má výraz `map head . snd`?

Jaký typ má `map head`? `[[a]] -> a`

Jaký typ má `snd`? `(b, c) -> c`

Jaký typ má `(.)`? `(e -> f) -> (d -> e) -> d -> f`

Algoritmické typování: příklad I

Jaký typ má výraz `map head . snd`?

Jaký typ má `map head`? $[[a]] \rightarrow a$

Jaký typ má `snd`? $(b, c) \rightarrow c$

Jaký typ má `(.)`? $(e \rightarrow f) \rightarrow (d \rightarrow e) \rightarrow d \rightarrow f$

$(.) :: (e \rightarrow f) \rightarrow (d \rightarrow e) \rightarrow d \rightarrow f$

`map head` :: $([[a]] \rightarrow [a])$

`snd` :: $((b, c) \rightarrow c)$

Algoritmické typování: příklad I

Jaký typ má výraz `map head . snd`?

Jaký typ má `map head`? $[[a]] \rightarrow a$

Jaký typ má `snd`? $(b, c) \rightarrow c$

Jaký typ má `(.)`? $(e \rightarrow f) \rightarrow (d \rightarrow e) \rightarrow d \rightarrow f$

Rovnosti:

$$(.) :: (e \rightarrow f) \rightarrow (d \rightarrow e) \rightarrow d \rightarrow f$$
$$\text{map head} :: ([[a]] \rightarrow [a])$$
$$\text{snd} :: ((b, c) \rightarrow c)$$

Algoritmické typování: příklad I

Jaký typ má výraz `map head . snd`?

Jaký typ má `map head`? $[[a]] \rightarrow a$

Jaký typ má `snd`? $(b, c) \rightarrow c$

Jaký typ má `(.)`? $(e \rightarrow f) \rightarrow (d \rightarrow e) \rightarrow d \rightarrow f$

Rovnosti: $e = [[a]]$, $f = [a]$

$(.) :: (e \rightarrow f) \rightarrow (d \rightarrow e) \rightarrow d \rightarrow f$

`map head` :: $([[a]] \rightarrow [a])$

`snd` :: $((b, c) \rightarrow c)$

Algoritmické typování: příklad I

Jaký typ má výraz `map head . snd`?

Jaký typ má `map head`? $[[a]] \rightarrow a$

Jaký typ má `snd`? $(b, c) \rightarrow c$

Jaký typ má `(.)`? $(e \rightarrow f) \rightarrow (d \rightarrow e) \rightarrow d \rightarrow f$

Rovnosti: $e = [[a]]$, $f = [a]$, $d = (b, c)$, $e = c$

$(.) :: (e \rightarrow f) \rightarrow (d \rightarrow e) \rightarrow d \rightarrow f$

`map head` :: $([[a]] \rightarrow [a])$

`snd` :: $((b, c) \rightarrow c)$

Algoritmické typování: příklad I

Jaký typ má výraz `map head . snd`?

Jaký typ má `map head`? $[[a]] \rightarrow a$

Jaký typ má `snd`? $(b, c) \rightarrow c$

Jaký typ má `(.)`? $(e \rightarrow f) \rightarrow (d \rightarrow e) \rightarrow d \rightarrow f$

Rovnosti: $e = [[a]]$, $f = [a]$, $d = (b,c)$, $e = c$

<code>(.)</code>	<code>::</code>	$(e \rightarrow f)$	<code>→</code>	$(d \rightarrow e)$	<code>→</code>	d	<code>→</code>	f
<code>(.)</code>	<code>::</code>	$([[a]] \rightarrow [a])$	<code>→</code>	$((b, [[a]]) \rightarrow [[a]])$	<code>→</code>	$(b, [[a]])$	<code>→</code>	$[a]$
<code>map head</code>	<code>::</code>	$([[a]] \rightarrow [a])$						
<code>snd</code>	<code>::</code>			$((b,c) \rightarrow c)$				

Algoritmické typování: příklad I

Jaký typ má výraz `map head . snd`?

Jaký typ má `map head`? $[[a]] \rightarrow a$

Jaký typ má `snd`? $(b, c) \rightarrow c$

Jaký typ má `(.)`? $(e \rightarrow f) \rightarrow (d \rightarrow e) \rightarrow d \rightarrow f$

Rovnosti: $e = [[a]]$, $f = [a]$, $d = (b, c)$, $e = c$

<code>(.)</code>	$::$	$(e \rightarrow f) \rightarrow$	$(d \rightarrow e) \rightarrow$	$d \rightarrow$	f
<code>(.)</code>	$::$	$([[a]] \rightarrow [a]) \rightarrow$	$((b, [[a]]) \rightarrow [[a]]) \rightarrow$	$(b, [[a]]) \rightarrow$	$[a]$
<code>map head</code>	$::$	$([[a]] \rightarrow [a])$			
<code>snd</code>	$::$		$((b, c) \rightarrow c)$		
<code>snd</code>	$::$		$((b, [[a]]) \rightarrow [[a]])$		

Algoritmické typování: příklad I

Jaký typ má výraz `map head . snd`?

Jaký typ má `map head`? $[[a]] \rightarrow a$

Jaký typ má `snd`? $(b, c) \rightarrow c$

Jaký typ má `(.)`? $(e \rightarrow f) \rightarrow (d \rightarrow e) \rightarrow d \rightarrow f$

Rovnosti: $e = [[a]]$, $f = [a]$, $d = (b, c)$, $e = c$

<code>(.)</code>	$::$	$(e \rightarrow f)$	\rightarrow	$(d \rightarrow e)$	\rightarrow	d	\rightarrow	f
<code>(.)</code>	$::$	$([[a]] \rightarrow [a])$	\rightarrow	$((b, [[a]]) \rightarrow [[a]])$	\rightarrow	$(b, [[a]])$	\rightarrow	$[a]$
<code>map head</code>	$::$	$([[a]] \rightarrow [a])$						
<code>snd</code>	$::$			$((b, c) \rightarrow c)$				
<code>snd</code>	$::$			$((b, [[a]]) \rightarrow [[a]])$				
<hr/>								
<code>result</code>	$::$					$(b, [[a]])$	\rightarrow	$[a]$

Algoritmické typování: příklad II

Jaký typ má výraz `filter ((4 >) . maximum)`?

Algoritmické typování: příklad II

Jaký typ má výraz `filter ((4 >) . maximum)`?

Jaký typ má `(4 >) . maximum`?

Jaký typ má `filter`?

Algoritmické typování: příklad II

Jaký typ má výraz `filter ((4 >) . maximum)`?

Jaký typ má `(4 >)` . maximum?

Jaký typ má `(4 >)`?

Jaký typ má maximum?

Jaký typ má `(.)`?

Jaký typ má `filter`?

Algoritmické typování: příklad II

Jaký typ má výraz `filter ((4 >) . maximum)`?

Jaký typ má `(4>)` . maximum?

Jaký typ má `(4>)`?

Jaký typ má `4`?

Jaký typ má `(>)`?

Jaký typ má `maximum`?

Jaký typ má `(.)`?

Jaký typ má `filter`?

Algoritmické typování: příklad II

Jaký typ má výraz `filter ((4 >) . maximum)`?

Jaký typ má `(4>)` . maximum?

Jaký typ má `(4>)`?

Jaký typ má `4`? (Num a) => a

Jaký typ má `(>)`?

Jaký typ má `maximum`?

Jaký typ má `(.)`?

Jaký typ má `filter`?

Algoritmické typování: příklad II

Jaký typ má výraz `filter ((4 >) . maximum)`?

Jaký typ má `(4>)` . maximum?

Jaký typ má `(4>)`?

Jaký typ má `4`? (Num a) => a

Jaký typ má `(>)`? (Ord b) => b -> b -> Bool

Jaký typ má maximum?

Jaký typ má `(.)`?

Jaký typ má filter?

Algoritmické typování: příklad II

Jaký typ má výraz `filter ((4 >) . maximum)`?

Jaký typ má `(4>) . maximum`?

Jaký typ má `(4>)`?

Jaký typ má `4`? `(Num a) => a`

Jaký typ má `(>)`? `(Ord b) => b -> b -> Bool`

`(>) :: (Ord b) => b -> b -> Bool`

`4 :: (Num a) => a`

Jaký typ má `maximum`?

Jaký typ má `(.)`?

Jaký typ má `filter`?

Algoritmické typování: příklad II

Jaký typ má výraz `filter ((4 >) . maximum)`?

Jaký typ má `(4>) . maximum`?

Jaký typ má `(4>)`?

Jaký typ má `4`? `(Num a) => a`

Jaký typ má `(>)`? `(Ord b) => b -> b -> Bool`

Rovnosti: `b = a`

`(>) :: (Ord b) => b -> b -> Bool`

`4 :: (Num a) => a`

Jaký typ má `maximum`?

Jaký typ má `(.)`?

Jaký typ má `filter`?

Algoritmické typování: příklad II

Jaký typ má výraz `filter ((4 >) . maximum)`?

Jaký typ má `(4>) . maximum`?

Jaký typ má `(4>)`?

Jaký typ má `4`? `(Num a) => a`

Jaký typ má `(>)`? `(Ord b) => b -> b -> Bool`

Rovnosti: `b = a`

`(>) :: (Ord b) => b -> b -> Bool`

`(>) :: (Ord a) => a -> a -> Bool`

`4 :: (Num a) => a`

Jaký typ má `maximum`?

Jaký typ má `(.)`?

Jaký typ má `filter`?

Algoritmické typování: příklad II

Jaký typ má výraz `filter ((4 >) . maximum)`?

Jaký typ má `(4>) . maximum`?

Jaký typ má `(4>)`?

Jaký typ má `4`? $(\text{Num } a) \Rightarrow a$

Jaký typ má `(>)`? $(\text{Ord } b) \Rightarrow b \rightarrow b \rightarrow \text{Bool}$

Rovnosti: $b = a$

$(>) \quad :: \quad (\text{Ord } b) \quad \Rightarrow \quad b \rightarrow b \rightarrow \text{Bool}$

$(>) \quad :: \quad (\text{Ord } a) \quad \Rightarrow \quad a \rightarrow a \rightarrow \text{Bool}$

$4 \quad \quad :: \quad (\text{Num } a) \quad \Rightarrow \quad a$

$(4>) \quad :: \quad (\text{Ord } a, \text{Num } a) \quad \Rightarrow \quad a \rightarrow \text{Bool}$

Jaký typ má `maximum`?

Jaký typ má `(.)`?

Jaký typ má `filter`?

Algoritmické typování: příklad II

Jaký typ má výraz `filter ((4 >) . maximum)`?

Jaký typ má `(4 >)` . maximum?

Jaký typ má `(4 >)`? (Ord a, Num a) => a -> Bool

Jaký typ má maximum?

Jaký typ má `(.)`?

Jaký typ má filter?

Algoritmické typování: příklad II

Jaký typ má výraz `filter ((4 >) . maximum)`?

Jaký typ má `(4 >)` . `maximum`?

Jaký typ má `(4 >)`? `(Ord a, Num a) => a -> Bool`

Jaký typ má `maximum`? `(Ord b) => [b] -> b`

Jaký typ má `(.)`?

Jaký typ má `filter`?

Algoritmické typování: příklad II

Jaký typ má výraz `filter ((4 >) . maximum)`?

Jaký typ má `(4 >)` . maximum?

Jaký typ má `(4 >)`? `(Ord a, Num a) => a -> Bool`

Jaký typ má maximum? `(Ord b) => [b] -> b`

Jaký typ má `(.)`? `(d -> e) -> (c -> d) -> c -> e`

Jaký typ má `filter`?

Algoritmické typování: příklad II

Jaký typ má výraz `filter ((4 >) . maximum)`?

Jaký typ má `(4 >) . maximum`?

Jaký typ má `(4 >)`? $(\text{Ord } a, \text{Num } a) \Rightarrow a \rightarrow \text{Bool}$

Jaký typ má `maximum`? $(\text{Ord } b) \Rightarrow [b] \rightarrow b$

Jaký typ má `(.)`? $(d \rightarrow e) \rightarrow (c \rightarrow d) \rightarrow c \rightarrow e$

$(.) :: (d \rightarrow e) \rightarrow (c \rightarrow d) \rightarrow c \rightarrow e$

$(4 >) :: (\text{Ord } a, \text{Num } a) \Rightarrow (a \rightarrow \text{Bool})$

$\text{maximum} :: (\text{Ord } b) \Rightarrow ([b] \rightarrow b)$

Jaký typ má `filter`?

Algoritmické typování: příklad II

Jaký typ má výraz `filter ((4 >) . maximum)`?

Jaký typ má `(4 >)` . maximum?

Jaký typ má `(4 >)`? $(\text{Ord } a, \text{Num } a) \Rightarrow a \rightarrow \text{Bool}$

Jaký typ má maximum? $(\text{Ord } b) \Rightarrow [b] \rightarrow b$

Jaký typ má `(.)`? $(d \rightarrow e) \rightarrow (c \rightarrow d) \rightarrow c \rightarrow e$

Rovnosti: $d = a, e = \text{Bool}$

$(.) :: (d \rightarrow e) \rightarrow (c \rightarrow d) \rightarrow c \rightarrow e$

$(4 >) :: (\text{Ord } a, \text{Num } a) \Rightarrow (a \rightarrow \text{Bool})$

$\text{maximum} :: (\text{Ord } b) \Rightarrow ([b] \rightarrow b)$

Jaký typ má `filter`?

Jaký typ má výraz `filter ((4 >) . maximum)`?

Jaký typ má `(4 >) . maximum`?

Jaký typ má `(4 >)`? $(\text{Ord } a, \text{Num } a) \Rightarrow a \rightarrow \text{Bool}$

Jaký typ má `maximum`? $(\text{Ord } b) \Rightarrow [b] \rightarrow b$

Jaký typ má `(.)`? $(d \rightarrow e) \rightarrow (c \rightarrow d) \rightarrow c \rightarrow e$

Rovnosti: $d = a$, $e = \text{Bool}$, $c = [b]$, $d = b$

$(.) :: (d \rightarrow e) \rightarrow (c \rightarrow d) \rightarrow c \rightarrow e$

$(4 >) :: (\text{Ord } a, \text{Num } a) \Rightarrow (a \rightarrow \text{Bool})$

$\text{maximum} :: (\text{Ord } b) \Rightarrow ([b] \rightarrow b)$

Jaký typ má `filter`?

Jaký typ má výraz `filter ((4 >) . maximum)`?

Jaký typ má `(4 >)` . maximum?

Jaký typ má `(4 >)`? $(\text{Ord } a, \text{Num } a) \Rightarrow a \rightarrow \text{Bool}$

Jaký typ má maximum? $(\text{Ord } b) \Rightarrow [b] \rightarrow b$

Jaký typ má `(.)`? $(d \rightarrow e) \rightarrow (c \rightarrow d) \rightarrow c \rightarrow e$

Rovnosti: $d = a$, $e = \text{Bool}$, $c = [b]$, $d = b$

$(.) :: (d \rightarrow e) \rightarrow (c \rightarrow d) \rightarrow c \rightarrow e$

$(.) :: (\text{Ord } a, \text{Num } a) \Rightarrow (a \rightarrow \text{Bool}) \rightarrow ([a] \rightarrow a) \rightarrow [a] \rightarrow \text{Bool}$

$(4 >) :: (\text{Ord } a, \text{Num } a) \Rightarrow (a \rightarrow \text{Bool})$

$\text{maximum} :: (\text{Ord } b) \Rightarrow ([b] \rightarrow b)$

Jaký typ má `filter`?

Jaký typ má výraz `filter ((4 >) . maximum)`?

Jaký typ má `(4>) . maximum`?

Jaký typ má `(4>)`? $(\text{Ord } a, \text{Num } a) \Rightarrow a \rightarrow \text{Bool}$

Jaký typ má `maximum`? $(\text{Ord } b) \Rightarrow [b] \rightarrow b$

Jaký typ má `(.)`? $(d \rightarrow e) \rightarrow (c \rightarrow d) \rightarrow c \rightarrow e$

Rovnosti: $d = a, e = \text{Bool}, c = [b], d = b$

$(.) :: (d \rightarrow e) \rightarrow (c \rightarrow d) \rightarrow c \rightarrow e$

$(.) :: (\text{Ord } a, \text{Num } a) \Rightarrow (a \rightarrow \text{Bool}) \rightarrow ([a] \rightarrow a) \rightarrow [a] \rightarrow \text{Bool}$

$(4>) :: (\text{Ord } a, \text{Num } a) \Rightarrow (a \rightarrow \text{Bool})$

$\text{maximum} :: (\text{Ord } b) \Rightarrow ([b] \rightarrow b)$

$\text{maximum} :: (\text{Ord } a, \text{Num } a) \Rightarrow ([a] \rightarrow a)$

Jaký typ má `filter`?

Jaký typ má výraz `filter ((4 >) . maximum)`?

Jaký typ má `(4>) . maximum`?

Jaký typ má `(4>)`? $(\text{Ord } a, \text{Num } a) \Rightarrow a \rightarrow \text{Bool}$

Jaký typ má `maximum`? $(\text{Ord } b) \Rightarrow [b] \rightarrow b$

Jaký typ má `(.)`? $(d \rightarrow e) \rightarrow (c \rightarrow d) \rightarrow c \rightarrow e$

Rovnosti: $d = a, e = \text{Bool}, c = [b], d = b$

$(.) :: (d \rightarrow e) \rightarrow (c \rightarrow d) \rightarrow c \rightarrow e$

$(.) :: (\text{Ord } a, \text{Num } a) \Rightarrow (a \rightarrow \text{Bool}) \rightarrow ([a] \rightarrow a) \rightarrow [a] \rightarrow \text{Bool}$

$(4>) :: (\text{Ord } a, \text{Num } a) \Rightarrow (a \rightarrow \text{Bool})$

$\text{maximum} :: (\text{Ord } b) \Rightarrow ([b] \rightarrow b)$

$\text{maximum} :: (\text{Ord } a, \text{Num } a) \Rightarrow ([a] \rightarrow a)$

$\text{result} :: (\text{Ord } a, \text{Num } a) \Rightarrow [a] \rightarrow \text{Bool}$

Jaký typ má `filter`?

Algoritmické typování: příklad II

Jaký typ má výraz `filter ((4 >) . maximum)`?

Jaký typ má `(4 >) . maximum`?

`(Ord a, Num a) => [a] -> Bool`

Jaký typ má `filter`?

Algoritmické typování: příklad II

Jaký typ má výraz `filter ((4 >) . maximum)`?

Jaký typ má `(4 >) . maximum`?

`(Ord a, Num a) => [a] -> Bool`

Jaký typ má `filter`? `(b -> Bool) -> [b] -> [b]`

Algoritmické typování: příklad II

Jaký typ má výraz `filter ((4 >) . maximum)`?

Jaký typ má `(4 >) . maximum`?

`(Ord a, Num a) => [a] -> Bool`

Jaký typ má `filter`? `(b -> Bool) -> [b] -> [b]`

`filter :: (b -> Bool) -> [b] -> [b]`

`(4 >).maximum :: (Ord a, Num a) => ([a] -> Bool)`

Algoritmické typování: příklad II

Jaký typ má výraz `filter ((4 >) . maximum)`?

Jaký typ má `(4 >) . maximum`?

`(Ord a, Num a) => [a] -> Bool`

Jaký typ má `filter`? `(b -> Bool) -> [b] -> [b]`

Rovnosti: `b = [a]`

`filter` :: `(b -> Bool) -> [b] -> [b]`

`(4 >).maximum :: (Ord a, Num a) => ([a] -> Bool)`

Algoritmické typování: příklad II

Jaký typ má výraz `filter ((4 >) . maximum)`?

Jaký typ má `(4 >) . maximum`?

`(Ord a, Num a) => [a] -> Bool`

Jaký typ má `filter`? `(b -> Bool) -> [b] -> [b]`

Rovnosti: `b = [a]`

`filter` :: `(b -> Bool) -> [b] -> [b]`

`filter` :: `(Ord a, Num a) => ([a] -> Bool) -> [[a]] -> [[a]]`

`(4 >).maximum` :: `(Ord a, Num a) => ([a] -> Bool)`

Algoritmické typování: příklad II

Jaký typ má výraz `filter ((4 >) . maximum)`?

Jaký typ má `(4 >) . maximum`?

`(Ord a, Num a) => [a] -> Bool`

Jaký typ má `filter`? `(b -> Bool) -> [b] -> [b]`

Rovnosti: `b = [a]`

`filter :: (b -> Bool) -> [b] -> [b]`

`filter :: (Ord a, Num a) => ([a] -> Bool) -> [[a]] -> [[a]]`

`(4 >).maximum :: (Ord a, Num a) => ([a] -> Bool)`

`result :: (Ord a, Num a) => [[a]] -> [[a]]`

Typování funkcí – lambda-abstrakce

Jaký typ má funkce: `(\x y -> takeWhile y ('a': x)) ?`

Typování funkcí – lambda-abstrakce

Jaký typ má funkce: $(\lambda x y \rightarrow \text{takeWhile } y ('a': x))$?

Vidíme, že funkce bere 2 argumenty (x a y) a vrací jednu hodnotu. Takže zatím máme typ: $a \rightarrow b \rightarrow c$.

Typování funkcí – lambda-abstrakce

Jaký typ má funkce: $(\lambda x y \rightarrow \text{takeWhile } y ('a': x))$?

Vidíme, že funkce bere 2 argumenty (x a y) a vrací jednu hodnotu. Takže zatím máme typ: $a \rightarrow b \rightarrow c$.

V těle lambda-abstrakce je x použito ve výrazu $('a': x)$, z toho vyplývá, že x musí být typu $[\text{Char}]$. Dostáváme $[\text{Char}] \rightarrow b \rightarrow c$.

Typování funkcí – lambda-abstrakce

Jaký typ má funkce: `(\x y -> takeWhile y ('a': x)) ?`

Vidíme, že funkce bere 2 argumenty (`x` a `y`) a vrátí jednu hodnotu. Takže zatím máme typ: `a -> b -> c`.

V těle lambda-abstrakce je `x` použito ve výrazu `('a': x)`, z toho vyplývá, že `x` musí být typu `[Char]`. Dostáváme `[Char] -> b -> c`.

Parametr `y` je použit jako první argument funkce `takeWhile` `:: (a -> Bool) -> [a] -> [a]`. Protože už víme, že `x` je typu `[Char]` a to je zároveň typ druhého argumentu `takeWhile`, vidíme, že typ `y` je `Char -> Bool`. Dostáváme `[Char] -> (Char -> Bool) -> c`.

Typování funkcí – lambda-abstrakce

Jaký typ má funkce: $(\lambda x y \rightarrow \text{takeWhile } y ('a': x))$?

Vidíme, že funkce bere 2 argumenty (x a y) a vrátí jednu hodnotu. Takže zatím máme typ: $a \rightarrow b \rightarrow c$.

V těle lambda-abstrakce je x použito ve výrazu $('a': x)$, z toho vyplývá, že x musí být typu $[\text{Char}]$. Dostáváme $[\text{Char}] \rightarrow b \rightarrow c$.

Parametr y je použit jako první argument funkce $\text{takeWhile} :: (a \rightarrow \text{Bool}) \rightarrow [a] \rightarrow [a]$. Protože už víme, že x je typu $[\text{Char}]$ a to je zároveň typ druhého argumentu takeWhile , vidíme, že typ y je $\text{Char} \rightarrow \text{Bool}$. Dostáváme $[\text{Char}] \rightarrow (\text{Char} \rightarrow \text{Bool}) \rightarrow c$.

Návratová hodnota funkce je stejná jako funkce takeWhile , čili $[a]$. Po naší specializaci je výsledkem

$$(\lambda x y \rightarrow \text{takeWhile } y ('a': x)) :: [\text{Char}] \rightarrow (\text{Char} \rightarrow \text{Bool}) \rightarrow [\text{Char}]$$