



Paradigmata programování 2  $\diamond$  poznámky k přednášce

## 10. Výpočty v $\lambda$ -kalkulu

verze z 5. května 2019

Obsah této přednášky najdete v prezentaci.

### Otázky a úkoly na cvičení

Úkoly se týkají vylepšené verze interpretu, která pracuje se jmény a kterou máte k dispozici.

1. Definujte logické spojky NOT, AND, OR.
2. Pomocí výrazu  $\leq$ , který je definován ve zdrojovém kódu interpretu definujte test na rovnost čísel:

$$= m n \rightarrow \text{FALSE}$$

pokud se čísla  $m$  a  $n$  nerovnají,

$$= n n \rightarrow \text{TRUE}$$

3. K práci se seznamy potřebujeme ještě prázdný seznam a predikát, který rozhoduje, zda je seznam prázdný. Navrhněte výrazy EMPTY a EMPTY? tak, aby

$$\text{EMPTY? EMPTY} \rightarrow \text{TRUE}$$

a pro libovolný pár  $P$

$$\text{EMPTY? } P \rightarrow \text{FALSE}$$

4. Lze vyřešit předchozí úlohu pro  $\text{EMPTY} \equiv \text{FALSE}$ ?
5. Definujte operaci rozdílu čísel. Při odečítání většího čísla od menšího může být výsledkem nula.
6. Redukce výrazu  $(+ 100 100)$  trvá v interpretu velice dlouho. Zkuste přijít na to, proč.

K řešení takových problémů je vhodné použít *profiler*. V LispWorks je jeden k dispozici.

7. Předpokládejme, že NOT je výraz, který realizuje negaci ve *zobecněné logice*, tedy

$$\text{NOT FALSE} \rightarrow \text{TRUE}$$

a

$$\text{NOT } M \rightarrow \text{FALSE}$$

pro libovolný výraz  $M$ , který se neredukuje na FALSE. Z vlastnosti Y-kombinátoru

$$(Y f) \rightarrow f (Y f)$$

dostáváme

$$(Y \text{NOT}) \rightarrow \text{NOT } (Y \text{NOT})$$

Co z toho plyne?