



# DATABÁZOVÉ A ZNALOSTNÍ IS

**P4****2008-03-13**

## $\lambda$ -KALKUL:

### Lambda kalkul:

- ✓ Autorem je americký matematik Alonzo Church (1936)
- ✓ Lambda-kalkul lze popsat jako jednoduchý nástroj k popisu formálních operací s výrazy, a nebo pro popis vlastností prvků množin. Je proto využitelný i k popisu činnosti počítače, který manipuluje s daty pomocí algoritmu.
- ✓ Lambda-kalkul je vyvrcholením snah matematiků a filosofů o formální a vypočitatelný model světa – od Pythagorejců ve starém Řecku přes matematiky 18. Století.
- ✓ Lambda-kalkul spolu s Turingovým strojem patří mezi nástroje, které měly bezprostřední vliv na vznik samočinných počítačů.
- ✓ Na lambda-kalkul lze nahlížet jako na univerzální a formálně přesný programovací jazyk. Na jeho principech jsou založeny všechny P. J.

**Princip  $\lambda$ -kalkulu** – je založen na třech pojmech:

#### ✓ **Abstrakce:**

5.8=40      Kolik stojí 5 koláčů, když jeden je za 8 Kč?

7.8=56      Kolik stojí 7 koláčů, když jeden je za 8 Kč?

...

Jak tedy určíme cenu pro libovolný počet koláčů?

Koláče.8      Koláče je abstrakce pro proměnlivý počet.

#### ✓ **Funkce**

#### ✓ **Aplikace funkce:**

Aplikace funkce je použití nějaké funkce pro případ konkrétní hodnoty:

(NAHRAĎ koláče DO koláče.8) HODNOTOU: 7

Vyvolá následující sled operací:

(NAHRAĎ koláče=7 DO koláče.8)

(7.8)

56

### **Zápis lambda-kalkulu:**

- ✓ Existuje několik druhů zápisu lambda-kalkulu i několik typů lambda-kalkulu. Zde budeme používat beztypový kalkul se zápisem upraveným pro potřeby objektového modelování.
- ✓ Dvě části oddělené svislíkem, nebo tlustou tečkou.

#### ✓ **Příklad:**

(NAHRAĎ koláče DO koláče.8) HODNOTOU 7

$(\lambda \text{ koláče} \mid \text{koláče.8}) \triangleleft 7$

↓      ↓      Aplikace hodnoty do lambda-výrazu  
Tělo obsahující výraz zapsaný běžným matematickým způsobem  
Hlavička obsahující volné proměnné (lambda-proměnné)

### **$\beta$ -redukce:**

- ✓ Beta-redukce je operace, při které se nahrazují volné lambda-proměnné konkrétními aplikovanými hodnotami a vzniká tak běžný matematický výraz, který lze dále zpracovat běžnými úpravami jako je například vytýkání před závorku, výpočet hodnot atd.

#### ✓ **Například:**

Výraz  $(\lambda \text{ koláče} \mid \text{koláče.8}) \triangleleft 7$

vyvolá následující beta-redukci  $(\lambda \text{ koláče} \mid \text{koláče.8}) \triangleleft 7 \Rightarrow (\lambda \text{ koláče}=7 \mid \text{koláče.8})$  (dvojitá šipka znamená, že se jeden výraz přepisuje na druhý) což se dále upraví jako  $(\lambda \text{ koláče}=7 \mid \text{koláče.8}) \Rightarrow 7.8=56$



### **α-konverze:**

- ✓ Alfa-konverze je operace, při které se přejmenovávají volné lambda-proměnné na jiné jméno. Toto přejmenování nemá vliv na vlastnosti výrazu a slouží jen k tomu, aby se předcházelo kolizím v pojmenování u složitějších výrazů.
- ✓ **Například:**  
Výraz  $(\lambda \text{ koláče} \mid \text{koláče}.8)$   
Lze konvertovat na  $(\lambda x \mid x.8)$   
Důkaz:  $(\lambda x \mid x.8) \triangleleft :7 \Rightarrow (\lambda x=7 \mid x.8) \Rightarrow (7.8) \Rightarrow 56$

### **λ-výraz jako data:**

- ✓ Mějme lambda-výraz pro výpočet množství koláčů pro dětský tábor, kde první proměnná je počet dětí, kde každé dítě sní 2 koláče, a druhá proměnná je počet koláčů, který sní vedoucí:  $(\lambda a \lambda b \mid 2a+b)$ .
- ✓ Tento výraz lze kombinovat s výrazem pro výpočet ceny nákupu:  $(\lambda x \mid x.8)$ .
- ✓ První funkci lze také použít přímo jako hodnotu pro druhou funkci:  
 $(\lambda x \mid x.8) \triangleleft : (\lambda a \lambda b \mid 2a+b)$   
 $\Rightarrow \lambda x = (\lambda a \lambda b \mid 2a+b) \mid x.8$   
 $\Rightarrow ((\lambda a \lambda b \mid 2a+b).8)$
- ✓ Tento výraz určuje cenu nákupu, i když ještě neznáme počet dětí a hladovost vedoucího. Může s ním být v počítači nakládáno jako s obyčejnými hodnotami, i když je funkcí.

### **Aplikace λ-kalkulu:**

- ✓ Objekty mají tu vlastnost, že dokáží reagovat na požadavky, které jsou jim posílány.
- ✓  $\text{Osoby} // (\lambda x \mid x \triangleleft \text{vek} > 60)$  – selekce, napravo od svíslítko buď true nebo false