

Posudek *oponenta* diplomové práce

Jméno a příjmení autora posudku: Mgr. Tomáš Křen

Jméno a příjmení autora práce: Bc. František Farka

Název práce: Maintainable type classes for Haskell

Vlastní text:

Cílem této diplomové práce bylo prozkoumat a navrhnout řešení problému v systému typových tříd programovacího jazyka Haskell. Tento problém spočívá v obtížné udržovatelnosti hierarchií typových tříd; při změnách v definicích typových tříd dochází k nežádoucím problémům se zpětnou kompatibilitou. K tomuto problému dochází zejména ve chvíli, kdy zjistíme, že by bylo vhodné, aby nějaká již používaná typová třída byla podtřídou jiné typové třídy (např. *Monad* není podtřídou *Applicative*, i když by měla být). Vhodné vyřešení tohoto problému by pomohlo k lepší praktické použitelnosti jazyka Haskell.

Kapitola 1 obsahuje popis jazyka Haskell, s hlavním důrazem na popis sémantiky typových tříd pomocí formálních inferenčních pravidel. V této poměrně dost formální kapitole bych ocenil více praktických příkladů, které by čtenáři usnadnili pochopení.

Kapitola 2 nás podrobněji seznamuje s problémem udržovatelnosti hierarchie typových tříd a jak je ho možné v současné chvíli obejít. Tento problém a argumenty volající po novém a lepším řešení pomocí jazykového rozšíření jsou vyjádřeny přehledně a srozumitelně. Kapitola 3 porovnává několik různých jazykových rozšíření určených k vyřešení daného problému a shrnuje jejich nevýhody. Na základě této analýzy pak autor navrhuje své řešení, založené hlavně na jazykovém rozšíření umožňujícím definovat defaultní implementaci instance nadtřídy uvnitř podtřídy. Toto rozšíření je pak podrobně popsáno v kapitole 4, včetně formální sémantiky pomocí inferenčních pravidel ve stylu první kapitoly.

Kapitola 5 diskutuje širší souvislosti zvoleného řešení a předkládá praktické příklady ilustrující výhodnou použitelnost navrženého rozšíření. Kapitola 6 popisuje proof-of-concept implementaci navrženého jazykového rozšíření pro kompilátor GHC, která je součástí práce a myslím, že splňuje svůj účel dobře.

Práce je celkově zdařilá. Oceňuji na ní hlavně její praktickou využitelnost pro komunitu jazyka Haskell a zvolenou míru formálnosti v popisu sémantiky navrženého rozšíření. Autor jasně specifikuje nedostatky konkurenčních řešení a v čem je jeho řešení lepší. Práce je napsána srozumitelně, dobrou angličtinou, s občasnými drobnými překlepy. Autor splnil zadání práce.

Do diskuze mám následující dotaz: V závěru píšete, že dalším krokem v implementaci navržených rozšíření je mimo jiné testování na rozsáhlé sadě testovacích příkladů. Zvažoval jste jaké metody pro testování složitějších testovacích příkladů by bylo vhodné použít? Myslíte, že by bylo vhodné takové příklady automatizovaně generovat?

Doporučení k obhajobě:

Z výše uvedených důvodů práci *doporučuji* k obhajobě.

Vynikající práce vhodná pro soutěž studentských prací	ANO <input type="checkbox"/>
---	------------------------------

Seznam soutěží studentských prací, viz <http://www.mff.cuni.cz/studium/bcmgr/prace/>

Pokud jste výše zaškrtnli ANO, zdůvodněte prosím svůj návrh, případně uveďte konkrétní soutěž, pro kterou je práce vhodná (rámeček lze nechat prázdný, pokud za dostatečné zdůvodnění považujete text posudku):

V Praze dne: 27.8.2014

Podpis: