

# Posudek oponenta bakalářské práce

Jméno a příjmení autora posudku: RNDr. Tomáš Valla, Ph.D.

Jméno a příjmení autora práce: Herbert Ullrich

Název práce User extensible graph database

## Text posudku

Úkolem Herberta Ullricha bylo přispět do již existující grafové databáze. Cílem bylo jednak vyřešit problém izomorfních duplikátů, druhá problém zjištění, zda pro jistou třídu grafů jsou již uloženy v databázi všechny její prvky.

## Hodnocení

V textové části práce nejprve autor srozumitelně vysvětlí, jak databáze funguje, k čemu slouží a zavede řešený problém. V druhé kapitole nastíní problematiku testování izomorfismu a uvádí přehled několika knihoven pro detekci izomorfismů. Kapitola 3 obsahuje výsledků několika testů výkonnosti zmíněných knihoven a popis uloženého formátu grafu. Pro problém detekce izomorfismů autor nejprve knihovnou `nauty` pro vstupní graf spočte jeho kanonický tvar a následně do databáze ukládá pouze tyto kanonické tvary. Ve čtvrté kapitole je navrženo řešení detekce úplnosti uložených grafových tříd. Kapitola 5 obsahuje techničtější diskuzi implementace s několika příklady kódu.

Text je výborně srozumitelný a se skvělou angličtinou, všemu jsem rozuměl, struktura práce je dobře navržena a dává smysl. Písemná práce je také mimořádně pečlivě napsaná, s perfektní typografií a téměř bez překlepů či chybek.

Nyní se dostáváme k mým výhradám.

Výkonnost existujících kanonizačních knihoven je dobře známa z jiných zdrojů, knihovna `nauty` vychází ve většině testů nejlépe, opakované testování výkonnosti mi tedy přijde zbytečné. Samotná realizace tedy obnáší pouze zavolání knihovny a následnou práci s kanonizovaným tvarem grafu. To mi přijde jako triviální změna. Mimochodem, na testovací instalaci jsem dostával chybové hlášky, pokud jsem zadal graf ve formátu `graph6` („Computing error: Graph not readable by nauty.“), fungovalo to pouze na formátu matice sousednosti.

Problém testování úplnosti uložených grafových tříd vyřešil autor značným zjednodušením celé problematiky. Každá třída je vždy parametrizovaná počtem vrcholů grafu a jednou jedinou zvolenou grafovou vlastností. Do zdrojových kódů je pak zadrátováno pro cca 9 vybraných vlastností počty grafů tuto vlastnost splňujících, převzatých z databáze OEIS.

Při vložení grafu do databáze se pomocí knihovny Sage určí sada cca 40 grafových vlastností (jestli to správně chápu, toto však není dílo autora), které se pak uloží jako atributy záznamu.

Těchto vlastností je však výrazně více, než autorem realizovaných 9. Autor do databáze přidává tabulku, kam počítá, kolik grafů se zadanou jednou vlastností se v databázi vyskytuje. Vypadá to také, že update tohoto counteru se neděje automaticky po vložení, nýbrž dávkově (ručně spouštěným?) skriptem.

Čekal bych, že autor přijde s poněkud větším příspěvkem. Ne vždy je totiž počet grafů splňujících určitou vlastnost znám přesně a jsou známy jen různě přesné odhady. Například by tedy šel implementovat nějaký ukazatel „tato třída není úplná“, „není jasné“, „je úplná“. To by ovšem znamenalo významnější studium literatury. Navíc ne vždy dává smysl každou vlastnost parametrizovat počtem vrcholů.

Ve zdrojových kódech není zdokumentováno, kdo pracoval na kterém modulu. Není tedy jasné, jaký je konkrétní přínos autora.

Návod k instalaci je dle mě nedostatečný a já jsem podle něj nedokázal systém zprovoznit. Dokumentace zdrojových kódů nemá valnou úroveň a měl jsem potíže se ve zdrojácích orientovat a pochopit, k čemu který modul slouží.

Celkově mi přijde, že příspěvek autora není příliš významný. Backend i frontend psal někdo jiný, grafové vlastnosti počítá při vložení grafu Sage, kanonický tvar `nauty`, v zásadě se příspěvek redukuje na implementaci tabulky s počtem výskytu grafů s jedinou vlastností, která se jednou za čas přepočítá.

Celkovou kvalitu a přínos práce tedy hodnotím jako podprůměrnou.

**Doporučuji tedy práci přijmout a hodnotit známkou C.**

### **Otázky k obhajobě**

Podporuje databáze také vymazání grafu? Pokud ne, co tomu brání?

Proč byla databáze navržena jen pro neorientované grafy? Lze ji rozšířit i na např. orientované grafy?

Jaká část kódu, který jste napsal, byla nejnáročnější a proč?

### **Doporučení k obhajobě**

Z výše uvedených důvodů práci *doporučuji* k obhajobě.

V Praze dne 14. 6. 2018

Podpis: