Posudek diplomové práce

Matematicko-fyzikální fakulta Univerzity Karlovy

Autor práce	Robert Husák
Název práce	Source Code Assertions Verification Using Backward Symbolic Execution
Rok odevzdání	2017
Studijní program	Informatika **Studijní obor** Softwarové a datové inženýrství

Autor posudku	Pavel Parízek **Role** Oponent
Pracoviště	Katedra distribuovaných a spolehlivých systémů

Text posudku:

Cílem práce bylo vytvořit snadno použitelný nástroj, který dokáže najít všechny sekvence instrukcí a vstupní hodnoty porušující specifikace chování ve formě assertions.
Autor práce zvolil metodu řízenou požadavky ("demand-driven") v tom smyslu, že vždycky zpracovává jednu assertion označenou uživatelem, a založenou na zpětném symbolickém vykonávání. Implementace nazvaná AskTheCode má formu rozšíření pro Microsoft Visual Studio a podporuje jazyk C#. Systém je rozdělen do tří modulů – transformace zdrojového kódu do grafu control-flow, vlastní symbolické vykonávání, a uživatelské rozhraní.
Významnou součástí práce je také experimentální porovnání s jinými nástroji. Experimenty jsou dobře zpracované a jejich výsledky důkladně popsané.
AskTheCode funguje spolehlivě a podle očekávání. Nezaznamenal jsem žádné nedostatky. Uživatelské rozhraní je docela povedené.

Text práce je napsán v anglickém jazyce. Obsahuje celkem velký počet gramatických a stylistických chyb. Vybrané pasáže jsou poměrně stručné a málo srozumitelné – například celá kapitola 2 a do velké míry také kapitola 3. Autor mohl některé koncepty a dílčí aspekty řešení vysvětlit mnohem podrobněji, a také ukázat jak souvisí s tématem práce. Často schází návaznost mezi jednotlivými odstavci a kapitolami. Musím ale zdůraznit, že text obsahuje všechny důležité informace, ale kvalita prezentace je nižší.

Dále mám tyto výhrady:
1) Chybí zdůvodnění, proč se autor rozhodl použít zpětné symbolické vykonávání (třeba na základě obsahu kapitol 2 a 3).
2) Nedostatečná diskuze, respektive analýza úlohy. Text nesděluje důvody pro rozhodnutí, která autor učinil během návrhu a implementace.
3) Kapitoly 5 a 6 nepopisují skoro vůbec, proč jsou některé důležité prvky systému navrženy a implementovány právě tím způsobem, který autor zvolil.
4) Velmi chybí srovnání s nástrojem SnuggleBug, který pracuje na velmi podobném principu. Detaily obsahuje tato publikace: S. Chandra et al. SnuggleBug: A Powerful Approach to Weakest Preconditions. PLDI 2009.

Moje otázky na autora práce:
1) Jaké další přístupy ke analýze chování programů jste zvažoval (a proč), když ve názvu práce už je „backward symbolic execution"?
2) Jaké obtížné problémy jste musel řešit? Z textu práce toto není příliš zřejmé.
3) Navrhnul a implementoval jste nějaké nízko-úrovňové optimalizace?
4) Proč není podporován heap (reference, objekty a jejich atributy)?

5) V kapitole 5 není vysvětlen způsob, jak je podporován inter-procedurální control flow, tedy přechody mezi CFG.

Na závěr chci vyzdvihnout, že implementace je dobře zpracována, ale text má slabší úroveň.

Práci doporučuji k obhajobě.

Práci nenavrhuji na zvláštní ocenění.

Pokud práci navrhujete na zvláštní ocenění (cena děkana apod.), prosím uveďte zde stručné zdůvodnění (vzniklé publikace, významnost tématu, inovativnost práce apod.).

Datum 25. 1. 2017 **Podpis**