

Testování a verifikace softwaru - Zkouška

Radek Mařík, *

January 3, 2018

1 Zkouška 01TVS a její hodnocení, platí od 1. 1. 2018

- Zkoušení mohou být jen ti studenti, kteří získali zápočet ze cvičení.
- Zkouška má dvě části, písemnou a ústní. Ústní část je vedena jako obhajoba písemné práce s případnými doplňujícími otázkami.
- Písemka se skládá také ze dvou částí.
 1. Ve první části se řeší dva příklady odvozené z příkladů řešených na cvičeních. Pro řešení příkladů je povoleno použít učebních materiálů. Na vypracování je poskytnuto 60 minut.
 2. V druhé části je potřeba odpovědět na 4 otázky dle okruhů uvedených níže. Nesmí se používat jakékoliv učební materiály. Na vypracování je poskytnuto 60 minut.
- Každá odpověď a řešení příkladů je ohodnoceno známkou. Výsledná známka zkoušky je určena podle součtu všech bodů získaných v rámci tohoto předmětu v daném semestru podle klasifikační stupnice ČVUT.

*marikr@fel.cvut.cz

2 Okruhy otázek

1. Kvalita - koncept, filosofie a systémy, UML

- (a) Statistika softwarových projektů.
- (b) Požadované vlastnosti softwarových procesů.
- (c) Znamé buggy historie, jejich shrnutí s hlediska testování. Typické chyby softwarových projektů.
- (d) Koncept kvality, proaktivita a reaktivita.
- (e) Definice testování softwaru.
- (f) Vztah mezi kvalitou pociťovanou zákazníkem a kvalitou měřenou producentem softwaru
- (g) Taguchiho přístup, ztrátová funkce.
- (h) Co (ne-)lze očekávat při automatizaci testovacího procesu.

2. Optimalizace testování

- (a) Princip párového testování.
- (b) Postup optimalizace metodou ortogonálních polí.
- (c) Postup optimalizace metodou latinských čtverců.
- (d) Vlastnosti latinských čtverců a jejich ortogonálních párů.

3. Klasická metodologie testování

- (a) Softwarová chyba, jejich distribuce.
- (b) Úrovně testování. Typologie testování.
- (c) Terminologie testování. Testovací plán, specifikace procedury, záznam testu.
- (d) UI chyby, chyby omezení, procesní chyby.
- (e) Chyby řízení, strukturální chyby, chyby požadavků.
- (f) Datové chyby, chyby kódu, chyby v manipulaci s pamětí.

4. Strukturované testování. Testování toku řízení

- (a) Obecný postup vytvoření testů pomocí grafů.
- (b) Kritéria pokrytí.
- (c) Základní typy testovacích modelů.

- (d) Testování toku řízení - metoda hlavních cest.
- (e) Testování datového toku - metoda du-cest.

5. Testování stavových automatů

- (a) Charakteristiky stavů z pohledu testování. Skryté stavy.
- (b) Postup návrhu testů.
- (c) Formální konstrukce testů.

6. Ověřování modelu

- (a) Princip a základní charakteristiky metod ověřování modelů.
- (b) Kripkeho struktura a její rozšíření.
- (c) Architektura systému UPPAAL a jeho základní vlastnosti.
- (d) Časový automat a jeho sémantika
- (e) Vysvětlete základní entity modelování v UPPAAL: synchronizace a její typy, pozice a jejich speciální vlastnosti, stráž, invariant.
- (f) Vysvětlete pojem dosažitelnosti, bezpečnosti a živosti a jak se tyto vlastnosti ověřují v systému UPPAAL.
- (g) Používání invariantů a stráží nad hodinami v systému UPPAAL.

7. Temporální logiky

- (a) Cesta výpočtu a pojem času.
- (b) CTL* logika a její temporální operátory.
- (c) CTL logika a její temporální operátory.
- (d) LTL logika a její temporální operátory.
- (e) Temporální logika systému UPPAAL.

8. Formální metody

- (a) Z notace - principy a základní vlastnosti.
- (b) PVS systém = principy a základní vlastnosti.
- (c) Alloy - prvky syntaxe jazyka, signatura, pole, relace, množinové operátory, relační operátory, fakta, funkce, predikáty, tvrzení, jak se provádí analýza specifikace.

9. Testování OO softwaru

- (a) Anomálie DU párů
- (b) Testování párových sekvencí

10. Spolehlivost softwaru

- (a) Základní metriky kvality softwaru
- (b) Statické modely spolehlivosti vývoje softwaru

11. Statistické testování softwaru

- (a) Propagace kovarianční matice - explicitní vztah.
- (b) Přímé propagování variance.
- (c) Přímé propagování min/max chyby.