Oponentský posudek diplomové práce

Bc. Michal Pochobradský - Integrace skriptovacích jazyků do C++ kódu

Předložená diplomová práce se zabývá tématikou integrace skriptovacího jazyka do existujícího C++ kódu. Student měl za úkol vybrat sadu skriptovacích jazyků a zpracovat analýzu jejich vhodnosti pro integraci do plánovače výpočetního centra CERIT-SC.

Samotný text práce lze hodnotit pouze jako nedostatečný. Nemá smysl se zde rozepisovat o četných gramatických a stylistických chybách, student si jich je nyní jistě vědom. Zkráceně řečeno, text práce by potřeboval důsledný přepis. Podobně trpí i formátování práce, od obsahu zabírajícím 3 stránky po velmi nevhodné zalamování bloků textu v rámci celé práce. S ohledem na kvalitu samotného textu se proto budu po zbytek mého posudku zabývat pouze obsahem práce.

Po krátkém úvodu se student věnuje představení jednotlivých kandidátů na integraci do plánovače výpočetního centra CERIT-SC. Zvolená množina jazyků je dle mého názoru reprezentativní a dostatečně obsáhlá. Samotný popis způsobů integrace jednotlivých jazyků do existujícího C++ kódu je poměrně těžce čitelný, za to ale můžou jak dříve zmiňované problémy tak samotný obsah této kapitoly (jde v podstatě o shrnutí několika referenčních příruček). Nicméně je zřejmé, že student si integraci jednotlivých jazyků opravdu nastudoval. Na konci té samé kapitoly se student (bohužel velmi stručně) věnuje obecným problémům s integrací skriptovacích jazyků a podpůrným nástrojům (generátory kódu a C++ rozhraní).

Následuje opět bohužel velmi stručná kapitola popisující vyhodnocení výkonu jednotlivých skriptovacích jazyků v kontextu integrace do plánovače výpočetního centra CERIT-SC. Ke zvoleným metrikám nemám výtky, bohužel bych se velmi rád dočetl víc detailů k analýze rozdílů mezi jednotlivými jazyky. Student u některých testů prohlašuje výsledky z analýz, které prý provedl, ty ale v textu nejsou vůbec zachyceny (například radikální rozdíl mezi V8 a ostatními jazyky při paměťových přenosech). Výsledky měření a zdrojové kódy jednotlivých experimentů jsou nicméně součástí práce a po jejich prohlédnutí musím konstatovat, že jejich kvalita je výrazně lepší než samotný text práce (důsledně zdokumentované a dobře čitelné).

Poslední dvě kapitoly se věnují samotnému jádru práce. Vyhodnocení vhodnosti jednotlivých jazyků a popis implementační části práce. Tyto kapitoly konečně odhalují množství práce, které student odvedl, i když některá tvrzení při vyhodnocení vhodnosti nemají díky problémům předchozích kapitol velkou důkazní

oporu. Zdrojové kódy přiložené k práci jsou opět velmi dobře čitelné a důsledně zdokumentované.

Jakožto interní oponent musím konstatovat, že text neodpovídá množství práce, kterou student odvedl. Student splnil zadání práce, ale zároveň ho i překonal funkčním prototypem integrace V8 Javascript do plánovače CERIT-SC (funkčnost prototypu byla ověřena v testovacím provozu).

Celkově zde vidím práci, kterou není možné ohodnotit jednou známkou. Kdybych nebyl s objemem práce, kterou student odvedl, seznámen ještě před přečtením textu, hodnotil bych práci jako nedostatečnou. Na druhou stranu pokud by byl text práce přepracován, nevidím žádnou překážku udělit hodnocení výborně. **Proto celkově navrhuji hodnocení D-E.**

K obhajobě práce mám na studenta následující otázky:

1. Vyřešil by problémy se správou paměti mezi C++ kódem a skriptovacím jazykem specificky implementovaný chytrý ukazatel?
2. Je problém s nízkou rychlostí samotného přenosu dat mezi C++ kódem a V8 implementací Javascriptu něco o čem se zmiňují i jiné zdroje?
3. Práce se zabývá primárně prvotní integrací skriptovacího jazyka do C++ kódu. Jak by vypadalo hodnocení vhodnosti v kontextu dlouhodobé udržitelnosti?

V Brně 20. 6. 2016
RNDr. Šimon Tóth