

# Posudek oponenta závěrečné práce

České vysoké učení technické v Praze

Fakulta informačních technologií

**Student:** Václav Motyka  
**Oponent práce:** RNDr. Ondřej Suchý, Ph.D.  
**Název práce:** Efektivní algoritmy pro řešení problému nalezení konvexní obálky v 3D  
**Obor:** Teoretická informatika

**Datum vytvoření:** 8. 2. 2018

<b>Hodnotící kritérium:</b>	<b>Způsob hodnocení - následující škálou 1 až 5:</b>
<b>1. Náročnost a další komentář k zadání</b>	<b>1=mimořádně náročné zadání, 2=náročnější zadání, 3=průměrně náročné zadání, 4=lehčí, ale ještě dostatečně náročné zadání, 5=nedostatečně náročné zadání</b>
<b>Popis kritéria:</b> Podrobněji charakterizujte diplomovou (bakalářskou) práci a její případné návaznosti na předchozí nebo běžící projekty. Dále posuďte, čím je zadání této ZP náročné. (U obtížnější ZP lze dále tolerovat některé nedostatky, které by u ZP standardní obtížnosti tolerovány nebyly; a naopak u jednoduché ZP mohou být zjištěné nedostatky hodnoceny přísněji.)	
<b>Komentář:</b> Zadání považuji za mírně náročnější. Algoritmy jsou netriviální a geometrie v prostoru vyžaduje určitou míru obezřetnosti. Pro analýzu a návrh paralelizace je pak potřeba algoritmy detailně pochopit.	
<b>Hodnotící kritérium:</b>	<b>Způsob hodnocení - následující škálou 1 až 4:</b>
<b>2. Splnění zadání</b>	<b>1=zadání splněno, 2=zadání splněno s menšími výhradami, 3=zadání splněno s většími výhradami, 4=zadání nesplněno</b>
<b>Popis kritéria:</b> Posuďte, zda předložená ZP splňuje zadání. V komentáři uveďte body zadání, které nebyly zcela splněny, případně rozšíření ZP oproti původnímu zadání. Nebylo-li zadání zcela splněno, pokuste se posoudit závažnost, dopady a případně i příčiny jednotlivých nedostatků.	
<b>Komentář:</b> Zadání bylo splněno s výhradami.  Algoritmy dle části 1) nastudovány byly, u popisu některých však chybí detaily a mnohdy je z popisu v práci těžké, ne-li nemožné, části algoritmů pochopit. U geometrických algoritmů je často velký rozdíl mezi geometrickou intuicí a skutečnou manipulací s datovými strukturami. Práce však tyto dva pohledy ignoruje a pokud už jsou uvedeny, nijak je nepropojuje.  Část 2) zadání je splněna, byly implementovány 4 algoritmy místo požadovaných alespoň 3, jeden navíc ve dvou verzích.  U paralelizace v OpenMP (část 3) zadání) v práci chybí analýza možností paralelizace, je vždy přímo navržena jedna taková možnost, bez posouzení výhod a nevýhod zvolené varianty, či jiných možností paralelizace. Navíc u algoritmů Jarvis March je paralelní verze značně zpomalována nevhodným způsobem kontroly duplicitních stěn. U inkrementálního algoritmu a algoritmu QuickHull se paralelizace omezuje na jejich použití jako podprocedury v algoritmu Divide and Conquer, pro který se jedna možnost paralelizace přirozeně nabízí. Tento je pak nastaven tak, aby vstup rozdělil na přesně tolik částí, kolik je k dispozici vláken. Paralelní verze Divide and Conquer a v důsledku tak i QuickHullu je však ve výsledku pomalejší než sekvenční.  Pro generování bodů (část 4)) je použito volání existujícího programu s omezenými možnostmi. Generování bodů vlastním programem nebylo implementováno. Z různých geometrických útvarů požadovaných zadáním jsou tedy dostupné koule a čtverec, generování lokálních shluků bodů není provedeno vůbec.  Výkonnost algoritmů byla skutečně porovnána na serveru Star, práce zmiňuje ověření správnosti s existujícím řešením (část 5) zadání), ovšem není vysvětleno, jak bylo provedeno.	
<b>Hodnotící kritérium:</b>	<b>Způsob hodnocení - následující škálou 1 až 4:</b>
<b>3. Rozsah písemné zprávy</b>	<b>1=splňuje požadavky, 2=splňuje požadavky s menšími výhradami, 3=splňuje požadavky s většími výhradami, 4=nesplňuje požadavky</b>
<b>Popis kritéria:</b> Zhodnotte přiměřenost rozsahu předložené ZP vzhledem k obsahu, tj. zda všechny části ZP jsou informačně bohaté a ZP neobsahuje zbytečné části.	
<b>Komentář:</b> Rozsah splňuje požadavky.	

<i>Hodnotící kritérium:</i>	<i>Způsob hodnocení - bodové hodnocení 0 až 100 bodů (známka A až F):</i>
<b>4. Věcná a logická úroveň práce</b>	<b>60 (D)</b>
<i>Popis kritéria:</i> Posuďte, zda předložená ZP je po věcné stránce v pořádku, případně vyskytují-li se v práci věcné chyby nebo nepřesnosti. Zhodnoťte dále logickou strukturu ZP, návaznosti jednotlivých kapitol a pochopitelnost textu pro čtenáře.	
<i>Komentář:</i> Členění práce na kapitoly je obstojné, jen některé části kapitol 3 a 4 by možná lépe zapadly do kapitoly 2 (např. sekce 3.2.2 a sekce 4.4).  Řada definic je značně vágní a, pokud by čtenář nevěděl, co daný pojem znamená, těžko by to z definice zjistil. Definice mnohdy neodpovídají zvyklostem v oboru a na Fakultě informačních technologií. Definici mnohostěnu chybí, ačkoliv by na všech místech, kde je použit, mohl být použit pojem polytop s totožným významem, který práce dříve zavedla. Podobně u definice kolineárních bodů není využita dřívější definice afinně závislých bodů. Pojem stěna se používá v jiném významu, než jak je definován, proto jsou některá tvrzení nesmyslná.  Na straně 24 se testuje, zda jsou dvě stěny spolu konvexní i pro stěny, které nesdílí žádnou hranu. Pojem je definován až o stránku dále a to pouze pro stěny sdílející hranu.  U pseudokódů chybí komentář, který by osvětlil jednotlivé kroky, které jsou často značně odlišné od textového popisu algoritmů.  U složitosti QuickHullu v průměrném případě není uvedeno v jakém smyslu je případ průměrný.	
<i>Hodnotící kritérium:</i>	<i>Způsob hodnocení - bodové hodnocení 0 až 100 bodů (známka A až F):</i>
<b>5. Formální úroveň práce</b>	<b>70 (C)</b>
<i>Popis kritéria:</i> Posuďte správnost používání formálních zápisů obsažených v práci. Posuďte typografickou a jazykovou stránku ZP, viz Směrnice děkana č. 26/2017, článek 3.	
<i>Komentář:</i> Práce obsahuje několik typografických prohřešků (např. řádka začínající čárkou, dvě samostatné reference hned za sebou, atd.).  Místo psaní "je uveden níže", "dále", apod. by bylo často lépe použít odkaz na příslušné plovoucí prostředí.  U obrázků s výsledky algoritmů chybí jednotky, v nichž je uváděn čas, žlutá barva ve výsledkových obrázcích je prakticky neviditelná.  Plovoucí prostředí jsou špatně nastavena, takže dochází k roztahování textu.	
<i>Hodnotící kritérium:</i>	<i>Způsob hodnocení - bodové hodnocení 0 až 100 bodů (známka A až F):</i>
<b>6. Práce se zdroji</b>	<b>80 (B)</b>
<i>Popis kritéria:</i> Vyjádřete se k aktivitě studenta při získávání a využívání studijních materiálů k řešení ZP. Charakterizujte výběr studijních pramenů. Posuďte, zda student využil všechny relevantní zdroje nebo zda se pokoušel řešit již vyřešené problémy. Ověřte, zda jsou všechny převzaté prvky řádně odlišeny od vlastních výsledků a úvah, zda nedošlo k porušení citační etiky a zda jsou bibliografické citace úplné a v souladu s citačními zvyklostmi a normami.	
<i>Komentář:</i> U reference [28] chybí jeden z autorů.  K některým tvrzením by se jistě dali najít relevantnější zdroje než online kurzy, či poznámky z přednášek.	
<i>Hodnotící kritérium:</i>	<i>Způsob hodnocení - bodové hodnocení 0 až 100 bodů (známka A až F):</i>
<b>7. Hodnocení výsledků, publikační výstupy a ocenění</b>	<b>49 (F)</b>
<i>Popis kritéria:</i> Vyjádřete se k úrovni dosažených hlavních výsledků ZP, např. k úrovni teoretických výsledků, nebo k úrovni a funkčnosti technického nebo programového vytvořeného řešení, apod. Případně také zhodnoťte, zda software nebo zdrojové texty, které nevytvořil sám student, byly v ZP použity v souladu s licenčními podmínkami a autorským právem. Popište případnou publikační činnost a získaná ocenění související s řešením této ZP.	

**Komentář:**

Zásadním bodem výstupu práce by měla být implementace algoritmů.

Implementace používá nevhodné datové struktury (vektor pro 3 souřadnice), které pravděpodobně limitují efektivitu výsledného programu.

Pro hledání duplicit se používá naivního porovnávání všech dvojic, což opět limituje efektivitu výsledného programu.

Některé výpočty se v kódu opakují, místo aby byla použita funkce daný výsledek počítající.

Pokud je paralelní verze pomalejší, než sekvenční, bylo by asi správné zvážit jiné možnosti paralelizace.

Pro lepší porovnání výsledků jednotlivých algoritmů by bylo třeba využít různorodější data, např alespoň body do 10% od povrchu koule. Je možné, že typy dat, na kterých bylo prováděno porovnání prostě více vyhovují QuickHullu.

*Hodnotící kritérium:*

*Způsob hodnocení - nehodnotí se*

### 8. Komentář o využitelnosti výsledků

*Popis kritéria:*

Uvedte, zda hlavní výsledky ZP rozšiřují již publikované známé výsledky a/nebo přinášející zcela nové poznatky. Uvedte možnosti využití výsledků ZP v praxi.

**Komentář:**

Vzhledem ke špatné kvalitě implementace nejspíše nejsou výsledky využitelné.

*Hodnotící kritérium:*

*Způsob hodnocení - nehodnotí se*

### 9. Otázky k obhajobě

*Popis kritéria:*

Uvedte případné dotazy, které by měl student zodpovědět při obhajobě ZP před komisí (body oddělte odřázkami).

**Otázky:**

Myslíte si, že kontejner vector představuje způsob uložení tří souřadnic daného bodu s optimálním výkonem?

Jak se při měření rychlosti jednotlivých algoritmů měřil čas a kolik měření se průměrovalo?

Proč byla použita zhruba dva a půl roku stará verze překladače GCC?

Proč jste odpovědi na předchozí tři otázky, které byly položeny už u předchozí obhajoby, nezpracoval do textu práce a případně do implementace?

*Hodnotící kritérium:*

*Způsob hodnocení - bodové hodnocení 0 až 100 bodů (známka A až F):*

### 10. Celkové hodnocení

49 (F)

*Popis kritéria:*

Shrňte stránky ZP studenta, které nejvíce ovlivnily Vaše celkové hodnocení. Celkové hodnocení **nesmí** být aritmetickým průměrem či jinou hodnotou vypočtenou z hodnocení v předchozích jednotlivých kritériích 1 až 9.

**Text hodnocení:**

Zadání bylo splněno jen s výhradami. Nebyly analyzovány možnosti paralelizace, generování bodů bylo pouze převzato, v důsledku toho nebylo možné řádně porovnat výkonnost algoritmů.

Definice jsou vágní. Vzhledem k tomu, že se jedná o práci z oboru Teoretické informatiky, nelze přehlédnout také chyby ve vyjádřeních týkajících se složitosti algoritmů, vyjádření týkající se optimálního výkonu univerzálních kontejnerů pro specifické použití, nebo naprosto nevhodný způsob hledání duplicit.

Kvalita výsledné implementace je i ze softwareového hlediska špatná.

Vzhledem k tomu, že student neopravil řadu chyb vytýkaných u předchozí obhajoby, nelze práci doporučit k obhajobě.

Podpis oponenta práce: