



Posudek oponenta závěrečné práce

| | |
|-----------------------------|--|
| Oponent práce: | Ing. Daniel Langr, Ph.D. |
| Student: | Matouš Bílek |
| Název práce: | Maticová kalkulačka pro výpočet vlastních čísel |
| Obor / specializace: | Webové a softwarové inženýrství, zaměření Softwarové inženýrství |
| Vytvořeno dne: | 6. června 2022 |

Hodnotící kritéria

1. Splnění zadání

- ▶ [1] zadání splněno
- [2] zadání splněno s menšími výhradami
- [3] zadání splněno s většími výhradami
- [4] zadání nesplněno

Práce se zabývá návrhem a tvorbou nástroje pro výpočet několika vlastních čísel řádkové matice za pomoci restartované Arnoldiho metody. Zadání práce bylo splněno, všechny její cíle byly dostatečně formulované a naplněné.

2. Písemná část práce

95 /100 (A)

Celkově je písemná část práce po obsahové i formální stránce velmi zdařilá a takřka jí není co vytknout. Text je psán srozumitelně, práce je logicky správně členěná a jednotlivá témata na sebe přirozeně navazují. Především oceňuji detailní a pochopitelný popis problematiky numerického řešení problému vlastních čísel a aplikované Arnoldiho metody. Přesto bych upozornil na několik nedostatků, většinou však pouze okrajového významu:

- 1) Text je občas příliš členěn na odstavce. V některých případech dokonce odstavce obsahují pouze jedinou větu. Je to nestandardní a působí to poněkud rušivě. Viz např. kapitola Úvod.
- 2) V kapitole Úvod je nastíněn obsah celé práce a jsou tam odkazy na další kapitoly. Tyto odkazy jsou ale nestandardně tvořeny pomocí názvů kapitol, které navíc začínají malými písmeny. Rovněž tyto odkazy jsou pouze textové (tj. nejedná se o "klikací" hyper-odkazy). Standardní řešení je odkazovat se na kapitoly pomocí jejich čísel (doporučuji zde příkaz `\autoref` z Latexového balíčku `hyperref`).

3) Definice trojúhelníkové matice (plus některé další definice) ve skutečnosti definují horní trojúhelníkovou (upper triangular) matici. Dolní trojúhelníková (lower triangular) matice je rovněž trojúhelníkovou maticí, ale podle uvedené definice tomu tak není.

4) Implementovaná metoda je iterační a u iteračních numerických metod nás často zajímá konvergence. Čtenář by proto možná uvítal grafy konvergence, tj. závislost chyby/reziduí na indexu iterace.

3. Nepísemná část, přílohy

95 /100 (A)

Neformální část práce, tj. především zdrojové kódy implementovaného nástroje, považuji za takřka ukázkovou. Zdrojový kód je velice dobře a přehledně členěn do příslušné a pochopitelné adresářové struktury, která je i zmíněna v textu práce (Příloha C). Vytvořené uživatelské rozhraní je přehledné a interaktivní. K neformální stránce mám jen několik málo poznámek:

1) Implementace v jazyce C++ za použití nejnovějšího vydaného standardu C++20 může v dnešní době omezovat využitelnost. Např. v mém případě jsem měl k dispozici linuxovou distribuci Ubuntu 20.04, která v základu obsahu překladač GCC ve verzi 9, a ten nemá podporu všech částí z C++20 potřebných pro překlad projektu. Pro instalaci jsem tudíž musel nainstalovat ne-výchozí verzi GCC 11.

2) Je vyžadován CMake minimální verze 3.19, což rovněž může omezovat použitelnost. Např. zmíněná moje distribuce obsahuje pouze CMake verze 3.16 a novější distribuce sama o sobě ani nenabízí.

4. Hodnocení výsledků, jejich využitelnost

80 /100 (B)

Osobně mě nenapadá nějaká praktická využitelnost práce a jejích výsledků. Zde by možná pomohlo, kdyby projekt nebyl prezentován jen ve formě výsledných programů, ale rovněž by definoval programové rozhraní (API), které by umožnilo integraci implementovaných metod do dalších programů. Na druhou stranu ale může práce sloužit jako dobrý úvod do problematiky hledání vlastních čísel řádkových matic a použitých numerických metod (konkrétně tedy Arnoldiho metody a variant s restartem). Na práci by určitě šlo navázat i formou jejího rozšíření např. o možnosti paralelizace programového řešení či o implementaci alternativních numerických metod pro hledání vlastních čísel.

Celkové hodnocení

95 /100 (A)

Celkově práci považuji za velmi zdařilou jak po obsahové, formální i neformální (programové) stránce. Veškeré požadované cíle byly splněny a prezentovány pochopitelnou formou.

Otázky k obhajobě

1) Umí kalkulačka vypočítat a uložit vlastní vektory, příslušné k zjištěným vlastním číslům? Pokud ne, jak složité by bylo takovou funkcionalitu dodělat?

2) Nakolik jsou vlastnosti přidané standardem C++20 klíčové pro implementaci? Dal by se zdrojový kód upravit i pro překladače podporující pouze starší standardy?

3) Z jakého důvodu je vyžadována verze programu CMake 3.19, kterou často neobsahují ani současně běžně používané linuxové distribuce?

Instrukce

Splnění zadání

Posudte, zda předložená ZP dostatečně a v souladu se zadáním obsahově vymezuje cíle, správně je formuluje a v dostatečné kvalitě naplňuje. V komentáři uveďte body zadání, které nebyly splněny, posudte závažnost, dopady a případně i příčiny jednotlivých nedostatků. Pokud zadání svou náročností vybočuje ze standardů pro daný typ práce nebo student případně vypracoval ZP nad rámec zadání, popište, jak se to projevilo na požadované kvalitě splnění zadání a jakým způsobem toto ovlivnilo výsledné hodnocení.

Písemná část práce

Zhodnoťte přiměřenost rozsahu předložené ZP vzhledem k obsahu, tj. zda všechny části ZP jsou informačně bohaté a ZP neobsahuje zbytečné části. Dále posudte, 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. Posudte správnost používání formálních zápisů obsažených v práci. Posudte typografickou a jazykovou stránku ZP, viz Směrnice děkana č. 52/2021, článek 3.

Posudte, zda student využil a správně citoval relevantní zdroje. Ověřte, zda jsou všechny převzaté prvky řádně odlišeny od vlastních výsledků, zda nedošlo k porušení citační etiky a zda jsou bibliografické citace úplné a v souladu s citačními zvyklostmi a normami. Zhodnoťte, zda převzatý software a jiná autorská díla, byly v ZP použity v souladu s licenčními podmínkami.

Nepísemná část, přílohy

Dle charakteru práce se případně vyjádřete k nepísemné části ZP. Například: SW dílo – kvalita vytvořeného programu a vhodnost a přiměřenost technologií, které byly využité od vývoje až po nasazení. HW – funkční vzorek – použité technologie a nástroje, Výzkumná a experimentální práce – opakovatelnost experimentů.

Hodnocení výsledků, jejich využitelnost

Dle charakteru práce zhodnoťte možnosti nasazení výsledků práce v praxi nebo uveďte, zda výsledky ZP rozšiřují již publikované známé výsledky nebo přinášející zcela nové poznatky.

Celkové hodnocení

Shrňte stránky ZP, které nejvíce ovlivnily Vaše celkové hodnocení. Celkové hodnocení nemusí být aritmetickým průměrem či jinou hodnotou vypočtenou z hodnocení v předchozích jednotlivých kritériích. Obecně platí, že bezvadně splněné zadání je hodnoceno klasifikačním stupněm A.