

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Gabrijel Okorn

**Primerjava vzporednega
programiranja v programskih jezikih
C, Julia in Golang**

DIPLOMSKO DELO

UNIVERZITETNI ŠTUDIJSKI PROGRAM
PRVE STOPNJE
RAČUNALNIŠTVO IN INFORMATIKA

MENTOR: doc dr. Boštjan Slivnik

Ljubljana, 2024

To delo je ponujeno pod licenco *Creative Commons Priznanje avtorstva-Deljenje pod enakimi pogoji 2.5 Slovenija* (ali novejšo različico). To pomeni, da se tako besedilo, slike, grafi in druge sestavine dela kot tudi rezultati diplomskega dela lahko prosto distribuirajo, reproducirajo, uporabljajo, priobčujejo javnosti in predelujejo, pod pogojem, da se jasno in vidno navede avtorja in naslov tega dela in da se v primeru spremembe, preoblikovanja ali uporabe tega dela v svojem delu, lahko distribuira predelava le pod licenco, ki je enaka tej. Podrobnosti licence so dostopne na spletni strani creativecommons.si ali na Inštitutu za intelektualno lastnino, Streliška 1, 1000 Ljubljana.



Izvorna koda diplomskega dela, njeni rezultati in v ta namen razvita programska oprema je ponujena pod licenco GNU General Public License, različica 3 (ali novejša). To pomeni, da se lahko prosto distribuira in/ali predeluje pod njenimi pogoji. Podrobnosti licence so dostopne na spletni strani <http://www.gnu.org/licenses/>.

Besedilo je oblikovano z urejevalnikom besedil L^AT_EX.

Kandidat: Gabrijel Okorn

Naslov: Primerjava vzporednega programiranja v programskih jezikih C, Julia in Golang

Vrsta naloge: Diplomaska naloga na univerzitetnem programu prve stopnje
Računalništvo in informatika

Mentor: doc. dr. Boštjan Slivnik

Opis:

Besedilo teme diplomskega dela študent prepiše iz študijskega informacijskega sistema, kamor ga je vnesel mentor. V nekaj stavkih bo opisal, kaj pričakuje od kandidatovega diplomskega dela. Kaj so cilji, kakšne metode naj uporabi, morda bo zapisal tudi ključno literaturo.

Title: Comparison of Parallel Programming in the Programming Languages C, Julia, and Golang

Description:

English version of opis.

Zahvaljujem se mentorju prof. dr. Boštjanu Slivniku za zagnanost in usmerjanje ob izdelavi diplomske naloge. Prav tako se zahvaljujem družini in prijateljem za nenehno podporo ob študiju.

Kazalo

Povzetek

Abstract

1	Uvod	1
1.1	Motivacija	1
1.2	Cilj raziskave	2
1.3	Kratek pregled poglavij	2
2	Programski jeziki	5
2.1	C	5
2.2	Julia	5
2.3	Golang	5
3	Uporabljena metodologija	7
3.1	Primerjanje vzporednega programiranja	7
3.2	Merjenje časa izvajanja	7
3.3	Metode uporabljene pri implementaciji zglednih primerov . . .	7
4	Analitična primerjava jezikov	9
4.1	OpenMP	9
4.2	Julia Tasks	9
4.3	Goroutines	9

5	Kvantitativna primerjava jezikov	11
5.1	Predstavitev problemov	11
5.2	Implementacija zaporednih programov	11
5.3	Implementacija vzporednih programov	12
5.4	Meritve	12
5.5	Vpliv funkcijskih klicev	12
6	Sklep	13
6.1	Primernost uporabe	13
6.2	Krivulja učenja	13
6.3	Učinkovitost	13
	Literatura	15

Seznam uporabljenih kratic

kratica	angleško	slovensko
NP	Nondeterministic Polynomial time	Nondeterministično polinomski čas
OpenMP API	Open Multi-Processing Application Programmable Interface	Odprta večprocesnost Programski vmesnik za aplikacije

Povzetek

Naslov: Primerjava vzporednega programiranja v programskih jezikih C, Julia in Golang

Avtor: Gabrijel Okorn

Diplomska naloga raziskuje in primerja vzporedno programiranje v programskih jezikih C, Julia in Golang. S temeljito analizo jezikovnih struktur in knjižnic smo ugotavljali, kateri jezik in pristop je najprimernejši za reševanje specifičnih problemov. Začeli smo z raziskovanjem vsakega jezika posebej. Nato smo jezike med seboj primerjali na dva načina. Analitično smo primerjali pristope enega jezika z ekvivalenti oziroma alternativami v drugih dveh jezikih. Potem smo na primerih lastnih implementacij programov izvedli časovne meritve, ki so omogočile kvantitativno primerjavo jezikov. Na koncu smo na podlagi rezultatov pripravili smernice za uporabo teh jezikov pri vzporednem programiranju.

Ključne besede: Vzporedno programiranje, C, Julia, Golang, OpenMP, Goroutine.

Abstract

Title: Comparison of Parallel Programming in the Programming Languages C, Julia, and Golang

Author: Gabrijel Okorn

The thesis investigates and compares parallel programming in the programming languages C, Julia, and Golang. Through a thorough analysis of language structures, synchronization mechanisms, and libraries, we determined which language and approach are most suitable for solving specific problems. We began with researching each language individually. Then, we compared the languages in two ways. First, we conducted an analytical comparison by comparing the approaches of one language with the equivalents or alternatives in the other two languages. Next, we performed time measurements on examples of our implementations, which allowed for a quantitative comparison of the languages. Finally, based on the results, we prepared guidelines for using these languages in parallel programming.

Keywords: Parallel programming, C, Julia, Golang, OpenMP, Goroutine.

Poglavje 1

Uvod

Dandanes veliko problemov rešujemo z računalniki, ker so hitri in učinkoviti. Kljub temu lahko naletimo na probleme, ki so tako obsežni, da jim niti računalniki niso kos. Ti problemi navadno sodijo v množico tako imenovanih NP-težkih problemov. Kadar je to mogoče, takšne probleme rešujemo tako, da jih razdelimo na manjše probleme, ki so med seboj neodvisni in le te rešujemo hkrati. Tako dobimo vzporedne programe, ki nam reševanje osnovnih problemov časovno razbremenijo. Pri tem moramo biti pozorni, da morajo biti rezultati vzporednih programov enaki rezultatom zaporednih programov[2].

1.1 Motivacija

Pisanje vzporednih programov je zahtevno in natančno delo. Programer mora podrobno razumeti obravnavani problem in izvajanje tovrstnih programov. Zaželeno je tudi, da se na jezik, ki ga uporablja, njegove knjižnice in orodja spozna. Motivacija ob izdelavi tega diplomskega dela je torej bralcu poglobiti razumevanje o delovanju vzporednih programov, napisanih v programskih jezikih C, Julia in Golang. Prav tako nam je v interesu programerjem olajšati izbiro programskega jezika, kadar so postavljeni pred izzive, ki

zahtevajo vzporedno programiranje.

1.2 Cilj raziskave

Ob reševanju problemov spoznamo, da so nekateri jeziki s svojimi pristopi bolj ali manj primerni od drugih. V različnih problemskih domenah imajo namreč svoje prednosti in slabosti, zato se optimalna izbira jezika lahko glede na obravnavani problem razlikuje. Poleg tega se lahko razlikujejo tudi naše zahteve in pričakovanja za rešitve. Te se med seboj razlikujejo glede na enostavnost implementacije, učinkovitost izvajanja, zanesljivost programa in podobno, kar zopet vpliva na izbiro jezika.

Očitno je izbira jezika in njegovih orodij pri vzporednem programiranju pomembna in ne nujno lahka. V tej diplomski nalogi si bomo zato prizadevali odpraviti nekatere dileme in raziskati naslednja področja:

1. **Osnovno razumevanje jezika** - todo
2. **Delovanje vzporednih programov** - todo
3. **Učinkovitost in enostavnost programskih jezikov** - todo
4. **Izbira programskega jezika** - todo
5. **Vpliv funkcijskih klicev na čas izvajanja programa** - todo

1.3 Kratek pregled poglavij

V uvodu smo si zadali razloge in cilje, ki jih želimo doseči s to diplomsko nalogo. V 2. poglavju bomo naredili pregled literature na tem področju. Spoznali se bomo s knjižnicami, pristopi in konstrukti za vsakega izmed jezikov. V 3. poglavju bomo nadaljevali s pregledom metodologije, ki je bila uporabljena v sklopu te raziskave. Tu se bomo poglobili v uporabljene metode za primerjavo programskih jezikov in za implementacije algoritmov, ki jih bomo srečali v 5. poglavju. V 4. poglavju bomo analitično primerjali vsak

jezik z vsakim. To pomeni, da si bomo za pristope, ki jih ponuja jezik pogledali ekvivalentne pristope oziroma alternative drugih dveh jezikov. Poskusili bomo najti podobnosti in razlike tako v pisanju kot izvajanju programov, pri čemer si bomo prizadevali te razlike čim boljše ovrednotiti. V 5. poglavju bomo zopet primerjali vsak jezik z vsakim, le da bomo tokrat sodbo prepustili konkretnim primerom in številkam. Za te primere bomo izvedli časovne meritve s katerimi bomo naredili kvantitativno primerjavo jezikov. Na podlagi prejšnjih poglavij bomo v zadnjem 6. poglavju zaobjeli pridobljene rezultate in naredili smotrne smernice za vzporedno programiranje v naštetih jezikih.

Poglavje 2

Programski jeziki

V tem poglavju si bomo od bliže pogledali vsakega izmed jezikov, pri čemer se bomo osredotočili na njihove lastnosti za vzporedno programiranje.

2.1 C

OpenMP ni nov računalniški jezik. Deluje v povezavi s standardnim Fortanom ali C/C++. Sestavljen je iz niza direktiv prevajalnika, ki opisujejo paralelizem v izvorni kodi, skupaj s podporno knjižnico podprogramov, ki so na voljo aplikacijam. Skupno so te direktive in knjižnične rutine formalno opisane z vmesnikom za programiranje aplikacij (API), ki je zdaj znan kot OpenMP[1].

2.2 Julia

Todo

2.3 Golang

Todo

Poglavje 3

Uporabljena metodologija

V tem poglavju se bomo seznanili z metodami, ki jih bomo uporabili tekom primerjanja programskih jezikov in implementacijah programov v naslednjih poglavjih. Metode bomo opisali in obrazložili zakaj jih uporabljamo.

3.1 Primerjanje vzporednega programiranja

3.2 Merjenje časa izvajanja

3.3 Metode uporabljene pri implementaciji zglednih primerov

Poglavje 4

Analitična primerjava jezikov

4.1 OpenMP

4.2 Julia Tasks

4.3 Goroutines

Poglavje 5

Kvantitativna primerjava jezikov

5.1 Predstavitev problemov

To do

5.1.1 Problem particij na eni veliki množici

5.1.2 Problem particij na več majhnih množicah

5.1.3 Algoritem Fruchterman Reingold

5.2 Implementacija zaporednih programov

To do

5.2.1 C

5.2.2 Julia

5.2.3 Golang

5.3 Implementacija vzporednih programov

To do

5.3.1 C

5.3.2 Julia

5.3.3 Golang

5.4 Meritve

To do

5.5 Vpliv funkcijskih klicev

To do

Poglavje 6

Sklep

6.1 Primernost uporabe

6.2 Krivulja učenja

6.3 Učinkovitost

Literatura

- [1] Rohit Chandra. *Parallel programming in OpenMP*. Morgan kaufmann, 2001.
- [2] kopl. *Kratek uvod v vzporedno programiranje*. URL: <https://hackmd.io/@hpcfs/wiki/%2FBN-ZX6AsSgqzmFMjwflVXQ%3Fview> (pridobljeno 31. 7. 2024).