Univerzitet u Beogradu - Elektrotehnički fakultet

Multiprocesorki sistemi (13S114MUPS, 13E114MUPS)



Domaći zadatak 1 – OPENMP

Izveštaj o urađenom domaćem zadatku

|  |  |
| --- | --- |
| Predmetni saradnici: | Studenti: |
| doc. dr Marko Mišić  dipl. ing. Pavle Divović | Uros Jovanovic 2018/0412 |

Beograd, novembar 2021.

Sadržaj

[Sadržaj 2](#_Toc529188776)

[1. Problem 1 - SGEMM 3](#_Toc529188777)

[1.1. Tekst problema 3](#_Toc529188778)

[1.2. Delovi koje treba paralelizovati 3](#_Toc529188779)

[1.2.1. Diskusija 3](#_Toc529188780)

[1.2.2. Način paralelizacije 3](#_Toc529188781)

[1.3. Rezultati 3](#_Toc529188782)

[1.3.1. Logovi izvršavanja 3](#_Toc529188783)

[1.3.2. Grafici ubrzanja 4](#_Toc529188784)

1. Problem 1 – Game of Life

U okviru ovog poglavlja je dat kratak izveštaj u vezi rešenja zadatog problema 1. Rešenja svih problema dati u okviru **jednog** dokumenta. U zavisnosti od broja postavljenih problema, dodati nova poglavlja u dokument.

* 1. Tekst problema

Paralelizovati program koji implementira simulaciju ćelijskog automata Game of Life. Simulacija je predstavljena dvodimenzionalnom matricom dimenzija w x h, a svaka ćelija c može uzeti vrednost 1 ukoliko predstavlja živu ćeliju, a 0 ukoliko je mrtva. Za svaku ćeliju se vrši izračunavanje vrednosti n koja predstavlja zbir živih ćelija u susedstvu posmatrane ćelije. Posmatra se osam suseda. Ćelije se rađaju i umiru prema pravilima iz sledeće tabele.

* 1. Delovi koje treba paralelizovati
     1. Diskusija

Ono gde moze da se primeni paralelizacija koda je sama funkcija evolve, gde uz pomoc openmp api-ja je moguce paralelizovati obilazak matrice i racunanje reziltata.

* + 1. Način paralelizacije

Samo racunanje rezultata za neku celiju je jedan posao koji moze da se izvrsava u paraleli sa ostalim racunanjima stanja celija. Kako rezultati ne bi bili korumpirani(odnosno da neke celije napreduju za nekoliko generacija brze od ostalih celija) koristi se pomocna matrica next\_generation gde se racuna medju rezultat i na kraju jedne generacije rezultati te matrice se prepisuju u matricu curr\_generation. To prepisivanje takodje moze da se paralelizuje.

* 1. Rezultati

U okviru ove sekcije su izloženi rezultati paralelizacije problema 1.

* + 1. Logovi izvršavanja

Ovde su dati logovi izvršavanja za definisane test primere i različit broj niti. Obavezno uključiti u ispis i vremena izvršavanja. Logove pojedinačno uokviriti i obeležiti.

Matrix dimension: 30x30

Iterations: 1000

Time elapsed: 29ms

-------------------------

Matrix dimension: 30x30

Iterations: 1000

Time elapsed: 37ms

Listing 1. Pralelno izvršavanje GAMEOFLIFE

Matrix dimension: 500x500

Iterations: 10

Time elapsed: 58ms

-------------------------

Matrix dimension: 500x500

Iterations: 10

Time elapsed: 72ms

Listing 2. Pralelno izvršavanje GAMEOFLIFE

Matrix dimension: 1000x1000

Iterations: 100

Time elapsed: 1037ms

--------------------------

Matrix dimension: 1000x1000

Iterations: 100

Time elapsed: 3041ms

Listing 3. Pralelno izvršavanje GAMEOFLIFE

Matrix dimension: 1000x1000

Iterations: 1000

Time elapsed: 10822ms

---------------------------

Matrix dimension: 1000x1000

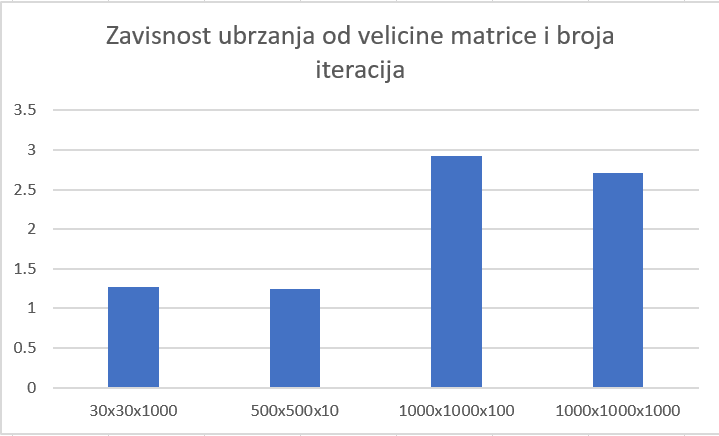
Iterations: 1000

Time elapsed: 29344ms

Listing 4. Pralelno izvršavanje GAMEOFLIFE

* + 1. Grafici ubrzanja

U okviru ove sekcije su dati grafici ubrzanja u odnosu na sekvencijalnu implementaciju.



Slika 1. Grafik zavisnosti ubrzanja naivnog algoritma od dužine teksta za N = 4 niti (primer)

* + 1. Diskusija dobijenih rezultata

Iz dobijenig rezultat vidimo da sa porastom broja iteracija i sa povecanjem uzorka znatno se povecava i ubrazanje. Ono na sta se gubi vreme to je prepisivanje matrice next\_generation u matricu curr\_generation.