
Elektrotehnički fakultet u Beogradu
Katedra za računarsku tehniku i informatiku

Predmet: Multiprocesorski sistemi (13S114MUPS)
Nastavnik: dr Milo Tomašević, red. prof.
Asistenti: doc. dr Marko Mišić; Pavle Divović, dipl. ing.
Ispitni rok: Treći kolokvijum (januar 2022.)
Datum: 25.01.2022.

Kandidat:* _____

Broj Indeksa:* _____

*Kolokvijum traje 105 minuta, prvih sat vremena nije dozvoljeno napuštanje kolokvijuma.
Upotreba literature nije dozvoljena.*

<i>Zadatak 1</i>	_____ /15	<i>Zadatak 5</i>	_____ /15
<i>Zadatak 2</i>	_____ /15	<i>Zadatak 6</i>	_____ /10
<i>Zadatak 3</i>	_____ /15	<i>Zadatak 7</i>	_____ /15
<i>Zadatak 4</i>	_____ /15		

Ukupno na kolokvijumu: _____ /100

Napomena: Ukoliko u zadatku nešto nije dovoljno precizno definisano, student treba da uvede razumnu pretpostavku, da je uokviri (da bi se lakše prepoznala prilikom ocenjivanja) i da nastavi da izgrađuje preostali deo svog odgovora na temeljima uvedene pretpostavke. Kod pitanja koja imaju ponuđene odgovore treba **samo zaokružiti** jedan odgovor. Na ostala pitanja odgovarati **čitko, kratko i precizno**.

* popunjava student.

3. [15] U kojim slučajevima nastaju problemi sa održavanjem inkluzije? Nacrtati i objasniti jedan primer narušavanja inkluzije u dvonivoskoj keš hijerarhiji.

4. [15] Kojoj grupi interkonekcionih mreža pripada mreža tipa k -arna d -kocka. Objasniti strukturu ove mreže i nacrtati je za $k = 3$ i $n = 81$? Napisati izraze za vrednosti uobičajenih parametara u opštem slučaju?

5. [15] U prilogu je dati CUDA jezgro za određivanje statistike pojavljivanja (histograma) mali slova engleskog alfabeza po podskupovima *a-d*, *e-h*, *i-l*, *m-p*, *q-t*, *u-x*, *y-z*. Kakve su performanse zadatog jezgra u smislu broja operacija u odnosu na broj pristupa globalnoj memoriji i šta predstavlja najveće usko grlo u tom smislu? Kojom tehnikom se ovo rešenje može poboljšati? Skicirati i diskutovati.

```
__global__ void histo_kernel(unsigned char *buffer,
                             long size, unsigned int *histo) {
    int i = threadIdx.x + blockIdx.x * blockDim.x;
    int stride = blockDim.x * gridDim.x;
    while (i < size) {
        int alphabet_position = buffer[i] - "a";
        if (alphabet_position >= 0 && alphabet_position < 26)
            atomicAdd(&histo[alphabet_position/4], 1);
        i += stride;
    }
}
```

6. [10] Definirati izvršni model CUDA arhitekture. Na koji način se elementi programskog modela preslikavaju na hardverske elemente grafičkog procesora? Nacrtati odgovarajuću sliku i objasniti.

7. [15] Koristeći CUDA tehnologiju paralelizovati deo koda koji rešava 2D Laplasovu jednačinu. Koristiti 2D organizaciju jezgra. Obratiti pažnju na efikasnost paralelizacije i koristiti deljenu memoriju, ukoliko je moguće. Smatrati da su podaci već inicijalizovani, a memorijski transferi izvršeni. Napisati poziv jezgra.

```
while ( error > tol && iter < iter_max ){
    error = 0.0;
    for( int j = 1; j < n-1; j++) {
        for( int i = 1; i < m-1; i++ )
        {
            Anew[j][i] = 0.25 * ( A[j][i+1] + A[j][i-1]
                                + A[j-1][i] + A[j+1][i]);

            double diff;
            if ((Anew[j][i] - A[j][i]) > 0.0) {
                diff = Anew[j][i] - A[j][i];
            }
            else {
                diff = A[j][i] - Anew[j][i];
            }
            if (diff > error) error = diff;
        }
    }
    for( int j = 1; j < n-1; j++){
        for( int i = 1; i < m-1; i++ ){
            A[j][i] = Anew[j][i];
        }
    }
    iter++;
}
```

