
Elektrotehnički fakultet u Beogradu
Katedra za računarsku tehniku i informatiku

Predmet: Multiprocesorski sistemi
(SI4MPS, IR4MPS, MS1MPS, 13E114MUPS)

Nastavnik: dr Milo Tomašević, vanr. prof.

Asistent: dipl. ing. Marko Mišić

Ispitni rok: februar 2015.

Datum: 10.02.2015.

Kandidat * : _____

Broj Indeksa * : _____

*Ispit traje 180 minuta, prvih sat vremena nije dozvoljeno napuštanje ispita.
Upotreba literature nije dozvoljena.*

<i>Zadatak 1</i>	_____ /5	<i>Zadatak 6</i>	_____ /10
<i>Zadatak 2</i>	_____ /10	<i>Zadatak 7</i>	_____ /10
<i>Zadatak 3</i>	_____ /15	<i>Zadatak 8</i>	_____ /10
<i>Zadatak 4</i>	_____ /10	<i>Zadatak 9</i>	_____ /15
<i>Zadatak 5</i>	_____ /5	<i>Zadatak 10</i>	_____ /10

Ukupno na ispitu: _____ /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.

1. [5] Objasniti probleme dovođenja energije i odvođenja toplote u savremenim procesorima.

2. [10] Objasniti karakteristike programskog modela *Data parallel*. Nacrtati i objasniti arhitekturu koja podržava ovaj model.

3. [15] Kod protokola MESI precizno:
- a) definisati stanja i transakcije na magistrali
 - b) nacrtati dijagram stanja i detaljno opisati sve akcije

4. [10] Objasniti šta je pravo deljenje, a šta lažno deljenje. Objasniti i diskutovati tehnike za smanjenje lažnog deljenja.

5. [5] Dati primer koji ilustruje narušavanje svojstva inkluzije u dvonivoskoj hijerarhiji keš memorija.

6. [10] Objasniti organizaciju kataloga kod *cache-based directory* protokola i opisati njegove osnovne akcije.

7. [10] Na koji način se može sprovesti operacija redukcije korišćenjem OpenMP tehnologije? Na primeru koda u prilogu, prikazati dva načina na koji se može obaviti ova operacija prilikom paralelizacije koda korišćenjem OpenMP.

```
for (j = 0; j < n; j++)  
    sum += b[j];
```

Rešenje:

I način	II način
<pre>for (j = 0; j < n; j++) sum += b[j];</pre>	<pre>for (j = 0; j < n; j++) sum += b[j];</pre>

8. [10] Šta predstavlja i za šta se može koristiti polje *tag* u rutinama za *point-to-point* komunikaciju MPI biblioteke? Da li je polje obavezno? Navesti primer rutine za prijem koja prihvata jedan ceo broj od procesa sa rangom 0 bez obzira na vrednost *tag*-a poruke.

9. [15] Napisati jezgro CUDA programa koje vrši obradu nad dvodimenzionalnom matricom celih brojeva. Jezgro treba da formira novu matricu na osnovu postojeće tako da svaki element novoformirane matrice dobija vrednost najvećeg od četiri moguća suseda (gore, dole, levo, desno). Smatrati da su matrice alocirane unapred. Koristiti 2D organizaciju jezgra. Prilikom rešavanja zadatka koristiti deljenu memoriju za smeštanje međurezultata i voditi računa da se ostvari maksimalan paralelizam. Navesti poziv (izvršnu konfiguraciju) jezgra za matricu dimenzija 500x200.

```
__global__ void maxmat (float* in, float* out, int m, int n);
```

10. [10] Dat je multiprocesorski sistem sa 4 identična procesora, koji koristi *Dragon* protokol za održavanje koherencije keš memorije. Svaka keš memorija ima po 2 ulaza, koji su veličine jedne reči. Preslikavanje je direktno. Početne vrednosti podataka su 0. Svaki upis uvećava vrednost izmenjenog podatka za 1. Na početku su sve keš memorije prazne. Data je sledeća sekvenca pristupa memoriji:

1. P0,R,A0	3. P1,R,A1	5. P0,R,A1	7. P0,W,A1
2. P1,W,A1	4. P0,W,A0	6. P2,W,A0	8. P1,R,A3

Napisati stanja koherencije u svim procesorima i stanje memorije posle svake promene i skicirati opisani sistem u trenutku 8. [8 poena]

Da li procesori pristupaju memoriji i kada? Za svaki pristup navesti razlog. [2 poena]

Trenutak 1												Memorija	
P0			P1			P2			P3			A0	

Pristupi memoriji:

Trenutak 2												Memorija	
P0			P1			P2			P3			A0	

Pristupi memoriji:

Trenutak 3												Memorija	
P0			P1			P2			P3			A0	

Pristupi memoriji:

Trenutak 4												Memorija	
P0			P1			P2			P3			A0	

Pristupi memoriji:

Trenutak 5

P0			P1			P2			P3		

Memorija	
A0	
A1	
A2	
A3	

Pristupi memoriji:

Trenutak 6

P0			P1			P2			P3		

Memorija	
A0	
A1	
A2	
A3	

Pristupi memoriji:

Trenutak 7

P0			P1			P2			P3		

Memorija	
A0	
A1	
A2	
A3	

Pristupi memoriji:

Trenutak 8

P0			P1			P2			P3		

Memorija	
A0	
A1	
A2	
A3	

Pristupi memoriji:
