
Elektrotehnički fakultet u Beogradu
Katedra za računarsku tehniku i informatiku

Predmet: Multiprocesorski sistemi (SI4MPS, IR4MPS, MS1MPS)

Nastavnik: dr Milo Tomašević, vanr. prof.

Asistent: dipl. ing. Marko Mišić

Ispitni rok: jun 2014.

Datum: 12.06.2014.

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>	_____ /5
<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>	_____ /10	<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 ponudene odgovore treba **samo zaokružiti** jedan odgovor. Na ostala pitanja odgovarati **čitko, kratko i precizno**.

* popunjava student.

1. [5] Objasniti kako povećanje broja procesora obično utiče na ubrzanje sistema i nacrtati tipičnu krivu. Objasniti razloge koji onemogućavaju da se postigne linearno ubrzanje.

2. [10] Definisti pojam sekvencijalne konzistencije. Neka se u nekom sekvencijalno konzistentnom sistemu izvršavaju procesi P1 i P2:

P1	P2
(1a) A = 1;	(2a) print B;
(1b) B = 2;	(2b) print A;

Pokazati da li je poredak izvršavanja 1b->1a->2b->2a ispravan?

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

4. [10] Objasniti 4C model promašaja u keš memoriji. Navesti načine kako se broj pojedinih vrsta promašaja može smanjiti.

5. [10] Objasniti kako se može smanjiti “visina” kataloga kod *directory* protokola, kao i osnovne operacije u takvom rešenju.

6. [5] Definisati pojam inkluzije u keš hijerarhiji. Objasniti osnovnu prednost koju primena inkluzije obezbeđuje.

7. [15] Korišćenjem OpenMP tehnologije, paralelizovati funkciju koja je data u prilogu. Funkcija vrši LU dekompoziciju kvadratne matrice koja je u memoriji linearizovana po vrstama. Obratiti pažnju na efikasnost paralelizacije.

```
void luDecomposition(float *a, int size) {
    int i,j,k;
    float sum;
    for (i=0; i<size; i++){
        for (j=i; j<size; j++){
            sum=a[i*size+j];
            for (k=0; k<i; k++) sum -= a[i*size+k]*a[k*size+j];
            a[i*size+j]=sum;
        }
        for (j=i+1;j<size; j++){
            sum=a[j*size+i];
            for (k=0; k<i; k++) sum -=a[j*size+k]*a[k*size+i];
            a[j*size+i]=sum/a[i*size+i];
        }
    }
}
```

Rešenje:

```
void luDecomposition(float *a, int size) {
    int i,j,k;
    float sum;

    for (i=0; i<size; i++){

        for (j=i; j<size; j++){
            sum=a[i*size+j];
            for (k=0; k<i; k++) sum -= a[i*size+k]*a[k*size+j];
            a[i*size+j]=sum;
        }

        for (j=i+1;j<size; j++){
            sum=a[j*size+i];
            for (k=0; k<i; k++) sum -=a[j*size+k]*a[k*size+i];
            a[j*size+i]=sum/a[i*size+i];
        }

    }
}
```

8. [10] Šta predstavlja pojam *memory coalescing*-a i na koji način niti treba da pristupaju podacima u globalnoj memoriji grafičkog procesora da bi se ostvarile maksimalne performanse?

9. [10] Šta radi i na koji način se može iskoristiti *split* operacija prilikom rada sa MPI procesima? Napisati deo koda koji sve procese iz MPI sveta deli u tri nezavisna komunikatora u zavisnosti od ranga konkretnog procesa.

10. [10] Dat je multiprocesorski sistem sa 4 identična procesora, koji koristi *MSI* 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,A0	5. P0,R,A2	7. P0,W,A2
2. P0,W,A0	4. P0,W,A0	6. P2,R,A1	8. P2,W,A1

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:
