
Elektrotehnički fakultet u Beogradu
Katedra za računarsku tehniku i informatiku

Predmet: Multiprocesorski sistemi (13E114MUPS, 13S114MUPS)

Nastavnik: dr Milo Tomašević, red. prof.

Asistent: doc. dr Marko Mišić; Matija Dodović, dipl. ing.

Ispitni rok: Avgust 2023.

Datum: 28.08.2023.

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>	_____ /10	<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 u okviru (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]** Ukratko objasniti na kojim nivoima paralelizma se istorijski zasnivalo poboljšanje performansi računara.

2. **[10]** Objasniti strukturu i osnovne karakteristike NUMA arhitektura. Diskutovati prednosti i nedostatke u odnosu na UMA arhitekturu.

3. **[10]**] Detaljno objasniti samo operacije upisa u protokolu MESI i nacrtati samo deo dijagrama stanja koji se na njih odnosi.

4. **[10]** Detaljno objasniti transakcije koje se odvijaju između dva nivoa hijerarhije keš memorija kada se primenjuje princip inkluzije.

5. [10] Objasniti strukturu kataloga kao i funkcionisanje *full-map directory* protokola. Uporediti njegove performanse sa performansama ostalih *directory* protokola.

6. [10] Objasniti i nacrtati interkonekcionu mrežu sa topologijom hiperkočke ($n = 4$). Navesti vrednosti relevantnih parametara. Koji je osnovni nedostatak?

7. [10] Neka se posmatra isečak koda u prilogu. Navesti gde u kodu postoje sinhronizacione tačke i objasniti da li se i na koji način može izvršiti optimizacija koda dostupnim naredbama OpenMP direktiva.

```
#pragma omp parallel
{
    #pragma omp for
    for (int i = 0; i < 10; ++i) c(i);
    #pragma omp single
    {
        d();
    }
    #pragma omp for
    for (int i = 0; i < 10; ++i) g(i);
}
```

8. [10] Priloženi kod predstavlja jednu naivnu implementaciju *broadcast* operacije korišćenjem MPI tehnologije. Na koji način se nedostaci implementacije mogu ispraviti korišćenjem rutina za asinhronu komunikaciju i zašto? Napisati odgovarajući deo koda koristeći rutine za asinhronu komunikaciju.

```
void my_bcast(void* data, int count, MPI_Datatype datatype,
              int root, MPI_Comm comm) {
    int world_rank, world_size;
    MPI_Comm_rank(comm, &world_rank);
    MPI_Comm_size(comm, &world_size);
    if (world_rank == root)
        for (int i = 0; i < world_size; i++)
            if (i != world_rank)
                MPI_Send(data, count, datatype, i, 0, comm);
    else
        MPI_Recv(data, count, datatype, root, 0, comm, MPI_STATUS_IGNORE);
}
```

9. [15] Korišćenjem CUDA tehnologije, paralelizovati kod u prilogu. Obratiti pažnju na efikasnost i korektnost paralelizacije.

```
void calc(float *x, float *y, float *work1, float *work2, int *ind,int n) {
    int i;
    for( i=0;i < n;i++) {
        x[i]= randPoint(); y[i]= randPoint();
        ind[i]= calcIndex(x[i], y[i]);
        work1[i]=i; work2[i]=i*i;
    }
    for( i=0;i< n;i++) {
        x[ind[i]] += work1[i];
        y[i] += work2[i];
    }
}
```

10. [10] Dat je multiprocesorski sistem sa 4 identična procesora, koji koristi *WTI* 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,W,A0	3. P2,R,A0	5. P2,W,A0	7. P1,R,A1
2. P1,W,A0	4. P0,W,A1	6. P2,R,A2	8. P1,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											
P0			P1			P2			P3		

Memorija	
A0	
A1	
A2	
A3	

Pristupi memoriji:

Trenutak 2											
P0			P1			P2			P3		

Memorija	
A0	
A1	
A2	
A3	

Pristupi memoriji:

Trenutak 3											
P0			P1			P2			P3		

Memorija	
A0	
A1	
A2	
A3	

Pristupi memoriji:

Trenutak 4											
P0			P1			P2			P3		

Memorija	
A0	
A1	
A2	
A3	

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:
