
Elektrotehnički fakultet u Beogradu
Katedra za računarsku tehniku i informatiku

Predmet: Multiprocesorski sistemi
(13E114MUPS, 13S114MUPS)

Nastavnik: dr Milo Tomašević, red. prof.

Asistent: dr Marko Mišić, dipl. ing.

Ispitni rok: jun 2017.

Datum: 05.06.2017.

Kandidat:* _____

Broj Indeksa:* _____

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

<i>Zadatak 1</i>	_____ /10	<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 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. [10] Objasniti šta je *power wall*. Kakve su posledice i implikacije ovog fenomena na projektovanje procesora? Navesti tehnike za smanjivanje snage.

2. [10] Objasniti koji se nivoi paralelizma i elementi *Flynn*-ove klasifikacije mogu pronaći u savremenim višejezgarnim procesorima.

3. [10] Objasniti kako se i u kojim situacijama vrši dinamička detekcija deljivosti kod MESI protokola. U kojem stanju može biti blok čija se ažurna kopija nalazi u samo jednom kešu? Objasniti.

4. [10] Objasniti tipove promašaja koji se javljaju u multiprocesorskim sistemima (4C model) i njihove uzroke. Objasniti tehnike za njihovo smanjivanje ili izbegavanje.

5. [10] Objasniti organizaciju i funkcionisanje kataloga kod *directory* šeme sa grubim vektorom. Ukoliko u sistemu postoji 256 procesora, a raspoloživa su četiri hardverska pokazivača po ulazu, koliko bitova ima svaki ulaz i kolika je veličina grupe?

6. [10] Objasniti motivaciju i princip organizacije *višestepenih interkonekcionih mreža* (MIN). Objasniti način povezivanja Omega mreže i nacrtati je za slučaj 8x8.

7. [10] CUDA jezgro u prilogu vrši operaciju redukcije nad elementima zadatog niza. Objasniti na koje se sve načine može izvršiti dodatna optimizacija priloženog koda, tako da se ubrza njegovo izvršavanje.

```
__global__ void reduce(int *g_idata, int *g_odata) {
    extern __shared__ int sdata[];
    unsigned int tid = threadIdx.x;
    unsigned int i = blockIdx.x * blockDim.x + threadIdx.x;
    sdata[tid] = g_idata[i];
    __syncthreads();
    for (unsigned int s = blockDim.x/2; s > 0; s >>= 1) {
        if (tid < s) sdata[tid] = sdata[tid] + sdata[tid + s];
        __syncthreads();
    }
    if (tid == 0) g_odata[blockIdx.x] = sdata[0];
}
```

8. [10] Objasniti u kojim situacijama je pogodno koristiti koncept *task*-ova prilikom paralelizacije korišćenjem OpenMP tehnologije? Navesti primer.

9. [15] Korišćenjem rutina iz MPI biblioteke, napisati deo koda koji vrši razmenu graničnih elemenata matrice unew. Smatrati da je matrica već ravnomerno raspodeljena procesima i da je MPI svet već inicijalizovan. Obratiti pažnju na efikasnost komunikacije.

```
int nx, ny, iterations;
double dx, dy, f[NX][NY], u[NX][NY], unew[NX][NY];
...
int i, it, j;
for ( it = 0; it < iterations; it++ ) {
    for ( j = 1; j < ny - 1; j++ ) {
        for ( i = 0; i < nx; i++ ) {
            unew[i][j] = 0.25 * (u[i-1][j] + u[i][j+1] + u[i][j-1] +
                                u[i+1][j] + f[i][j] * dx * dy );
        }
    }
    /* exchange border elements */
    memcpy(u, unew, ny * nx * sizeof(double));
}
```

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