
Elektrotehnički fakultet u Beogradu
Katedra za računarsku tehniku i informatiku

Predmet: Multiprocesorski sistemi
(13E114MUPS, 13S114MUPS, SI4MPS, IR4MPS, MS1MPS)
Nastavnik: dr Milo Tomašević, red. prof.
Asistent: dipl. ing. Marko Mišić
Ispitni rok: Septembar 2016.
Datum: 30.08.2016.

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 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] Nabrojati osnovne klase i nivoe paralelizma.

2. [10] Nacrtati i objasniti strukturu NUMA generičke arhitekture. Kako ona može da se prilagođava različitim programskim modelima?

3. [10] Objasniti i obrazložiti potrebu za adaptivnim protokolima. Objasniti njihov uobičajeni način rada.

4. [10] Objasniti 4C model promašaja u multiprocesorskim sistemima. Objasniti kako se može smanjiti broj ovih tipova promašaja u keš memorijama.

5. [10] Nacrtati i objasniti strukturu jednog ulaza kataloga $Dir_3 B$ protokola. Objasniti osnovne operacije. Koji su nedostaci protokola?

6. [10] Nacrtati opštu strukturnu šemu dvonivoskog hijerarhijskog sistema sa distribuiranom memorijom zasnovanog na magistralama. Objasniti strukturu i princip rada.

7. [10] U prilogu je dat primer jednog CUDA jezgra koje vrši uparivanje stringova. Diskutovati aritmetički intenzitet jezgra i njegovu pogodnost za izvršavanje na grafičkom procesoru i predložiti eventualna poboljšanja. Funkcija *atomicAdd* je ugrađena CUDA funkcija za atomično dodavanje.

```
__global__ void kernel (int* tokA, int* tokB, int* match, int* matchIndex){
int idx = blockIdx.x * blockDim.x + threadIdx.x;
int idy = blockIdx.y * blockDim.y + threadIdx.y;
int substrLen = 0;
if (tokA.h[idx] == tokB.h[idy])    {
    while ((idx + substrLen < tokA.size) &&
           (idy + substrLen < tokB.size) &&
           (tokA.vector[idx + substrLen] ==
            tokB.vector[idy + substrLen]))
        substrLen++;
    if (substrLen >= MML) match[atomicAdd(matchIndex, 1)] = substrLen;
}
}
```

8. [10] Kakva je uloga *tag* polja u MPI *point-to-point* komunikaciji i u kojim situacijama ono može da pomogne? Da li proces može da primi poruku za bilo kojim *tag*-om i kako? Navesti primer.

9. [15] Korišćenjem OpenMP tehnologije, paralelizovati funkciju koja je data u prilogu. Funkcija vrši numeričku integraciju funkcije $f(x)$ na intervalu od a do b u n tačaka. Paralelizaciju izvršiti korišćenjem *worksharing* direktiva i korišćenjem taskova. Obratiti pažnju na efikasnost paralelizacije. Kratko diskutovati performanse oba rešenja.

```
for ( i = 0; i < n; i++ ) {  
    x = ( ( double ) ( n - i - 1 ) * a +  
          ( double ) ( i ) * b ) / ( double ) ( n - 1 );  
    total = total + f ( x );  
}
```

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. P0,W,A0	5. P2,R,A2	7. P1,W,A1
2. P1,W,A0	4. P0,W,A2	6. P1,R,A1	8. P0,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												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:
