
Elektrotehnički fakultet u Beogradu
Katedra za računarsku tehniku i informatiku

Predmet: Multiprocesorski sistemi (13S114MUPS)

Nastavnik: dr Milo Tomašević, red. prof.

Asistent: doc. dr Marko Mišić

Ispitni rok: Prvi kolokvijum (Novembar 2020.)

Datum: 03.11.2020.

Kandidat:* _____

Broj Indeksa:* _____

*Kolokvijum traje 105 minuta, prvih sat vremena nije dozvoljeno napuštanje kolokvijuma.
Upotreba literature nije dozvoljena.*

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

Ukupno na kolokvijumu: _____ /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. [15] Objasniti promenu principa projektovanja procesora koji favorizuju paralelno procesiranje.

2. [15] Šta je pretpostavka *Gustafson*-ovog zakona? Izvesti i precizno objasniti ovaj zakon.

3. [15] Opisati tehnološke trendove u procesorskoj tehnologiji i njihove posledice.

4. [15] Dati uporedni pregled karakteristika programskih modela zajedničke memorije i prenosa poruka. U čemu se razlikuju arhitekture i organizacije sistema koji direktno podržavaju ove modele.

5. [15] Korišćenjem OpenMP tehnologije, paralelizovati funkciju u prilogu koji računa vrednost određenog integrala korišćenjem Simpsonovog 1/3 pravila. Obratiti pažnju na efikasnost i korektnost paralelizacije. Smatrati da su sve promenljive ispravno deklarisanе.

```
float simpsons(float ll, float ul, int n) {

    float h = (ul - ll) / n, x[10], fx[10];

    for (int i = 0; i <= n; i++) {

        x[i] = ll + i * h; fx[i] = func(x[i]);

    }

    float res = 0;

    for (int i = 0; i <= n; i++) {

        if (i == 0 || i == n) res += fx[i];

        else if (i % 2 != 0) res += 4 * fx[i];

        else res += 2 * fx[i];

    }

    res = res * (h / 3);

    return res;

}
```

6. [10] Da li i na koji način OpenMP podržava ugneždeni paralelizam? Da li je moguće u okviru paralelnog regiona pokrenuti novi paralelni region i pod kojim uslovima?

7. [15] Neka se posmatra jedna aplikacija za prikupljanje i agregaciju podataka sa interneta. Aplikacija se sastoji od veb tragača (*web crawler*) koji prikuplja određene informacije na internetu, a zatim se radi obrada. Nakon merenja performansi sekvencijalne implementacije posmatrane aplikacije pri uobičajenoj upotrebi, dobijeni su sledeći rezultati: aplikacija 45% vremena provodi obavljajući ulazno-izlazne operacije (sekvencijalno prikupljanje podataka iz jednog izvora), a 55% vremena provodi u obradi podataka. Tipično vreme obrade jednog paketa podataka korišćenjem jednog jezgra je 1s.

a) [7] Ukoliko se aplikacija paralelizuje za izvršavanje na SMP sistemu sa 8 jezgara na 2GHz sa 32GB memorije, navesti formulu za Amdalov zakon i odrediti maksimalno moguće ubrzanje koje se može postići za zadatu aplikaciju.

b) **[8]** Da li i na koji način može organizovati paralelno izvršavanje sekvencijalnog dela aplikacije, ukoliko paketi sa interneta mogu nezavisno da se obrađuju? Smatrati da paketi imaju neujednačeno vreme obrade i diskutovati moguće načine paralelne obrade.