
Elektrotehnički fakultet u Beogradu
Katedra za računarsku tehniku i informatiku

Predmet: Multiprocesorski sistemi (13S114MUPS)

Nastavnik: dr Marko Mišić, vanr. prof.

Asistent: Matija Dodović, mast. ing.

Ispitni rok: Prvi kolokvijum (decembar 2025.)

Datum: 03.12.2025.

Kandidat:* _____

Broj Indeksa:* _____

*Kolokvijum traje 90 minuta. Prvih 60 minuta 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>	_____/15
<i>Zadatak 3</i>	_____/15	<i>Zadatak 7</i>	_____/10
<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.

- 2/6

3. **[15]** Navesti glavne trendove u razvoju memorijskih tehnologija i forme paralelizma koje se javljaju. Da li i na koji način oni prate razvoj procesora u ovom smislu? Objasniti.
4. **[15]** Na koje sve načine se može sprovoditi operacija redukcije kod programskih modela zajedničke memorije? Objasniti i ilustrovati na primeru sumiranja niza $A[1000]$ na multiprocesorskom sistemu sa 4 procesora.

5. [15] Korišćenjem OpenMP tehnologije i koncepta *task*-ova, paralelizovati kod koji je dat u prilogu. Kod vrši određene proračune u okviru stabla jedne *nbody* simulacije. Obratiti pažnju na efikasnost paralelizacije. Smatrati da su sve promenljive već definisane i inicijalizovane.

```
double calculate_mass(struct node_t *node) {

    if (!node->has_particle) {

        node->total_mass = 0;

    }

    else if (!node->has_children)      {

        node->total_mass = mass[node->particle];

    }

    else {

        node->total_mass = 0;

        for (int i = 0; i < 4; i++){

            node->total_mass += calculate_mass(&node->children[i]);

        }

    }

    return node->total_mass;
}

int main () {

    ...

    calculate_mass(root);

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

        sumx += mass[i] * x[i];

        sumy += mass[i] * y[i];

        total_mass += mass[i];

    }

    double cx = sumx / total_mass;

    double cy = sumy / total_mass;

}
```

6. **[15]** Direktive za podelu posla kod OpenMP tehnologije

- a. **[5]** Definirati pojam *master* i *single* direktive kod OpenMP tehnologije. Navesti prednosti i ograničenja.

- b. **[10]** Na primeru koda sa slike, upotrebiti *single* direktivu, tako da se deo koda koji učitava podatke radi samo od strane jedne niti. Objasniti kako se obezbeđuje da ostale niti ne nastave svoje izvršavanje pre nego što je podatak učitao.

```
#include <stdio.h>
#include <omp.h>

int main() {
    int data = 0;

    #pragma omp parallel default(shared)
    {
        int tid = omp_get_thread_num();

        printf("Thread %d is working in the parallel region.\n", tid);

        read_data(&data);

        printf("Thread %d proceeds. Shared data value is: %d\n", tid, data);

    }
    return 0;
}
```

7. **[10]** Definirati pojam dekompozicije domena prilikom podele posla u paralelnom programiranju i navesti moguće varijante. Odgovor ilustrovati slikom. Na koje sve načine je moguće izvršiti dekompoziciju domena korišćenjem OpenMP tehnologije?