

# Multiprocesorski sistemi

## Pismeni ispit, 14.09.2009.

Literatura nije dozvoljena.  
Ispit traje 180 minuta.

1. Definirati sekvencijalnu konzistenciju i dovoljne uslove da bi se ona garantovala. [5 poena]
2. Objasniti karakteristike *Dataflow* modela i arhitekture koja ga podržava i ilustrovati ga slikom. Koji problemi se javljaju kod ovog pristupa? [10 poena]
3. Objasniti 4C model promašaja u keš memoriji. Navesti načine kako se broj pojedinih vrsta promašaja može smanjiti. [5 poena]
4. Za *full-map directory* protokol objasniti organizaciju informacije o koherenciji, transakcije u sistemu i akcije protokola. Nacrtati i objasniti dijagram stanja i prelaza. [15 poena]
5. Objasniti logiku i strategiju adaptivnih protokola. Ukratko opisati stanja i akcije RWB protokola? Diskutovati postavljeni invalidacioni prag. [10 poena]
6. Objasniti kakve su prednosti organizacije čvora kao manjeg multiprocesora u hijerarhijskim sistemima. [5 poena]
7. Sastaviti Pthreads program na jeziku C ili C++ koji ispisuje srednju vrednost i standardnu devijaciju dinamičkog niza celih brojeva. Svaka stvorena nit treba da obradi isti broj elemenata. Glavni program upravlja nitima i obavlja svu komunikaciju sa korisnikom. Ako broj elemenata niza nije odgovarajući broju niti, prekinuti program. [15 poena]

$$\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i \quad \sigma = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^2}$$

8. Dat je multiprocesorski sistem sa 4 identična procesora, koji koristi MESI za održavanje koherencije keš memorije. Svaka keš memorija ima po 2 ulaza, koji su veličine jedne reči. Preslikavanje je asocijativno. Na početku su sve keš memorije prazne. Svaki upis povećava vrednost izmenjenog podatka za 1. Početne vrednosti podataka u memoriji su 0. Data je sledeća sekvenca pristupa memoriji:

1. P0, R, A0	3. P1, R, A2	5. P1, W, A0	7. P0, R, A0
2. P1, R, A0	4. P1, W, A2	6. P1, W, A0	8. P1, W, A0

- 8.1. Koliko puta koji od procesora pristupa memoriji? Za svaki pristup navesti razlog. [3 poena]
- 8.2. Koliki je Hit Rate za svaki od procesora (brojati i čitanje i upis, prikazati zbirno)? [3 poena]
- 8.3. Napisati stanja koherencije u svim procesorima i stanje memorije nakon svakog koraka (dovoljno je stanja ispisivati samo posle promene). [9 poena]
9. Sastaviti MPI program na jeziku C ili C++ koji vrši kvadriranje svih elemenata matrice celih brojeva. Svaki proces treba da obradi tačno po jednu vrstu matrice. Proces sa rangom 0 treba da obavi i unos ulaznih podataka i ispis dobijenih rezultata. Prilikom razmene podataka, procesi treba da jednu vrstu šalju samo jednom naredbom za slanje, u kojoj je broj podataka postavljen na 1. Ako dimenzije matrice nisu odgovarajuće broju procesa, prekinuti program. [20 poena]

### Napomena:

U zadacima pretpostaviti da funkcije koje obavljaju potrebne ulazne i izlazne radnje već postoje, tako da za njih samo treba navesti prototipove i pozvati ih na odgovarajućim mestima u programskom kodu. Pretpostaviti da korisnik unosi sintaksno ispravne podatke.

Ukoliko u bilo kom pitanju ili zadatku nešto nije dovoljno precizno definisano, student treba da uvede razumnu pretpostavku, da je u okviru (da bi bila lakše prepoznata prilikom ocenjivanja) i da nastavi da izgrađuje preostali deo svog odgovora na temeljima uvedene pretpostavke.