

# Multiprocesorski sistemi

## Pismeni ispit, 23.06.2008.

Literatura nije dozvoljena.  
Ispit traje 210 minuta.

1. Navesti osnovne karakteristike NUMA sistema. Dati sliku tipičnog sistema. [5 poena]
2. Koje su osnovne prednosti hardverske realizacije protokola za koherenciju keš memorija? [5 poena]
3. Objasniti 4C model promašaja u keš memoriji. Navesti načine kako se broj pojedinih vrsta promašaja može smanjiti. [10 poena]
4. Za protokol MESI objasniti stanja, transakcije na magistrali i akcije protokola. Nacrtati i objasniti dijagram stanja i prelaza. [15 poena]
5. Šta se postiže primenom "cache-based" protokola? Objasniti organizaciju kataloga i izračunati njegovu veličinu. Koje su prednosti i nedostaci? [10 poena]
6. Ukratko opisati *Dir; B* šemu. Koje su prednosti i nedostaci? [5 poena]
7. Koristeći POSIX niti napisati program na jeziku C ili C++ koji računa vrednost broja  $n$ . Proračun obaviti po sledećem: kružnica poluprečnika 1 je upisana u centar koordinatnog sistema; u gornjem desnom kvadrantu upisan je kvadrat stranice dužine 1, čije je levo donje teme u koordinatnom početku, stranice su paralelne apscisi, odnosno ordinati. Svaka od niti treba da prvo pseudoslučajno napravi koordinate  $N$  tačaka koje pripadaju opisanom kvadratu. Zatim, svaka od niti treba da prebroji koliko tačaka u kvadratu istovremeno pripada i datom krugu. Glavni program treba da na osnovu odnosa rezultata ova dva prebrojavanja izračuna vrednost broja  $n$ . [20 poena]
8. Dat je multiprocesorski sistem sa 4 identična procesora, koji koristi *Dragon* za održavanje koherencije keš memorije. Svaka keš memorija ima po 2 ulaza, koji su veličine jedne reči. Preslikavanje je asocijativno. Algoritam zamene je FIFO. Data je sledeća sekvenca pristupa memoriji:

1. P0,R,A0	3. P1,W,A2	5. P2,W,A0	7. P0,R,A0
2. P1,R,A0	4. P1,R,A1	6. P1,W,A0	8. P1,W,A0

Na početku su sve keš memorije prazne. Inicijalna vrednost svih memorijskih lokacija je 0. Instrukcija upisa povećava vrednost podatka za 1.

- 8.1. Skicirati opisani sistem posle trenutka 8. [3 poena]
  - 8.2. Koliko puta koji od procesora pristupa memoriji? [2 poena]
  - 8.3. Koliki je Hit Rate za svaki od procesora (brojati i čitanje i upis)? [2 poena]
  - 8.4. Napisati stanja koherencije u svim procesorima (samo posle promene) [8 poena].
9. Sastaviti MPI program na jeziku C ili C++ koji učitava, kvadrira i ispisuje sve elemente dinamičkog niza celih brojeva. Proces sa rangom 0 treba da obavlja svu komunikaciju sa korisnikom. [15 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 uokviri (da bi bila lakše prepoznata prilikom ocenjivanja) i da nastavi da izgrađuje preostali deo svog odgovora na temeljima uvedene pretpostavke.