

Multiprocesorski sistemi (SI4MPS)

Drugi kolokvijum, 01.12.2009.

Literatura nije dozvoljena.
Kolokvijum traje 90 minuta.

1. Opisati sistolne arhitekture i objasniti njihov princip rada. Koje su njihove prednosti i koje aplikacije efikasno podržavaju. [15 poena]
2. Objasniti prednosti i nedostatke korišćenja zajedničke keš memorije. [10 poena]
3. Definisati sekvencijalnu konzistenciju. Navesti dovoljne uslove da bi se ona garantovala. [10 poena]
4. Kod protokola *MSI* precizno objasniti: a) stanja, b) transakcije na magistrali i c) akcije protokola. Nacrtati dijagram stanja i prelaza. Navesti osnovne neefikasnosti ovog protokola. [25 poena]
5. 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 direktno. Početne vrednosti podataka su 0. Na početku su sve keš memorije prazne. Data je sledeća sekvenca pristupa memoriji:

| | | | |
|--------------|---------------------|---------------------|---------------|
| 1. P0, R, A0 | 4. P2, W(10hex), A0 | 7. P1, W(02hex), A0 | 10. P3, R, A2 |
| 2. P1, R, A0 | 5. P2, W(20hex), A0 | 8. P3, R, A0 | 11. P3, R, A3 |
| 3. P3, R, A1 | 6. P0, W(11hex), A0 | 9. P3, R, A1 | 12. P1, R, A2 |

- 5.1. Napisati stanja koherencije u svim procesorima (samo posle promene). [8 poena]
- 5.2. Koliko puta koji od procesora pristupa memoriji? Za svaki pristup navesti razlog. [4 poena]
- 5.3. Koliki je Hit Rate za svaki od procesora (brojati i čitanje i upis, prikazati zbirno)? [4 poena]
- 5.4. Skicirati opisani sistem posle trenutka 12. [4 poena]
6. Napisati program na programskom jeziku C ili C++ koji prvo učitava dimenzije, a potom i elemente niza realnih brojeva. Potrebno je u datom nizu pronaći elemente sa najmanjom i najvećom apsolutnom vrednošću. Obradu paralelizovati i ostvariti korišćenjem MPI. Proces sa rangom 0 učitava podatke sa standardnog ulaza, nakon čega ravnopravno učestvuje u poslu sa ostalim procesima i ispisuje rezultate pretrage (vrednosti i indekse elemenata) na standardnom izlazu. Pretpostaviti da je broj procesa uvek veći od 1. Ako korisnik unese broj elemenata niza koji nije celobrojni umnožak broja 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 uokviri (da bi bila lakše prepoznata prilikom ocenjivanja) i da nastavi da izgrađuje preostali deo svog odgovora na temeljima uvedene pretpostavke.