

1. Kako bi ubrzanje bilo veće od 1, trajanje komunikacije i čekanje mora biti kraće od trajanja računanja na pojedinom procesoru – TOČNO
2. Ukupno trajanje računanja T_R paralelnog programa može ovisiti o raspodjeli zadataka po procesorima. TOČNO
3. Uz povećanje količine računanja i nepromijenjene ostale elemente trajanje, učinkovitost programa pada. NETOČNO
4. Trajanje izvođenja paralelnog programa ne ovisi o promatranom procesoru iz skupa svih procesora koji izvode paralelni program. TOČNO
5. Sitnozrnata podjela posla podrazumijeva malu količinu komunikacije u odnosu na veću količinu računanja. NETOČNO
6. trajanje izvođenja paralelnog programa ovisi o promatranom procesoru iz skupa svih procesora. NETOČNO
7. Prilikom pridruživanja zadataka procesorima, zadatke koji se izvode neovisno poželjno je pridružiti istom procesoru. NETOČNO
8. Jednom procesoru može biti dodijeljeno više MPI procesa. TOČNO
9. Trajanje izvođenja paralelnog programa je po definiciji neovisno o promatranom procesoru. TOČNO
10. Uz smanjenje količine računanja i nepromijenjene ostale elemente trajanja računanja, učinkovitost programa raste. TOČNO
11. Kako bi ubrzanje bilo veće od 1 trajanje komunikacije i čekanja mora biti kraće od računanja na pojedinom procesu. TOČNO
12. Trajanje računanja T_r paralelnog programa može ovisiti o raspodjeli zadataka po procesorima. TOČNO

1. Ukupan broj poruka koje se razmijene (pošalju) u provedbi komunikacijske strukture binarnog stabla za 2D procesora iznosi $2d - 1$.
2. Kompozicija modula u paralelnim programima može biti: **Serijska, paralelna i zajednička kompozicija**.
3. Ako je učinkovitost 25%, a ubrzanje je 4, koliki je broj procesora? **16 ($S=P \cdot E$ $S=\text{ubrzanje}$, $E=\text{učinkovitost}$ pa onda $P=S/E$ tj $4/0.25 = 16$).**
4. Izoučinkovitost opisuje kako se mora promijeniti br procesora u ovisnosti o količini posla (računanja) kako bi **učinkovitost** ostala nepromijenjena.
5. Pridruživanje se provodi ukoliko je broj **zadataka** veći od broja **procesora**.
6. Na APRAM računalu, unutar istog asinkronog odsječka, samo **jedan** procesor smije pristupiti **istoj globalnoj** memorijskoj lokaciji.
7. Koje dodatne parametre uvodi model APRAM u odnosu na model PRAM? **B (vrijeme potrebno za sinkronizaciju p procesora) i d (vrijeme za globalno čitanje/pisanje).**

8. Ukupan broj poruka koji se razmjene(pošalju) u provedbi komunikacijske strukture hiperkocke za 2d procesa iznosi **$n \log n$** .
9. Povećanje zrnatosti može se postići tehnikama **Povećanje zadataka, uvišestručavanje računanja**.
10. Ako je ubrzanje linearno, učinkovitost je(kakvog iznosa?) **jednaka 1**.
11. Ako je superlinearno učinkovitost je veća od 1
12. Ako je sublinearno učinkovitost je manja od 1
13. Uz superlinearno ubrzanje, učinkovitost je (kakvog iznosa?) **>1**.
14. Navedite sve četiri vrste aPRAM instrukcija: Globalno čitanje, **Globalno pisanje, Lokalna operacija, Sinkronizacija / ograda**.
15. Funkcija izoučinkovitosti opisuje kako se mora promijeniti **količina posla** u ovisnosti o promjeni broja procesora kako bi **učinkovitost** ostala nepromijenjena.
16. Faza pridruživanja se provodi ukoliko je broj **Zadataka** veći od broja **procesora**.
17. Uvišestručavanje računanja je tehnika kojom se **povećava** ukupna količina računanja kako bi se **smanjila** količina komunikacije.
18. Pojava zagušenja voditelja u modelu voditelj-radnik moguća je uz **prevelik** broj radnika.
19. Ukoliko se ubrzanje paralelnog programa mjeri u odnosu na najbolji slijedni program, radi se o **apsolutnom** ubrzanju.
20. Ukoliko se ubrzanje paralelnog programa mjeri u odnosu na isti program pokrenut na jednom procesoru, radi se o **relativnom** ubrzanju.
21. MPI mehanizam dijeljenja komunikatora omogućava izvedbu **paralelne** kompozicije modula u paralelnom programu.
22. MPI mehanizam modula u paralelnim programima omogućava izvedbu **slijedne i paralelne kompozicije modula**.
23. Ubrzanje veće od linearnog naziva se **superlinearno**.
24. Povećanje zrnatosti možemo ostvariti tehnikama: **Povećanja zadataka i uvišestručavanja računanja**.
25. Prilikom istodobnog čitanja iste memorijske lokacije u CRCW PRAM računalu, svaki procesor će pročitati **istu/jednaku** vrijednost.
26. Izraz koji opisuje trajanje slanja jedne poruke duljine L riječi u jednostavnom modelu komunikacije je: **$T_{msg} = t_s + twL$ (t_s - postavljanje poruke, tw - prijenos jedne riječi).**
27. Zrnatost zadataka se može definirati kao **količine računanja (lokalnog rada) i količine komunikacije (nelokalnog rada)**.
28. U modelu raspodijeljene memorije, procesori mogu komunicirati jedino **razmjenom poruka**.
29. Složenost provedbe postupka scan niza duljine n elemenata na PRAM računalu uz p procesora gdje je $p < (n/2)$ iznosi **$O(n/p + \log p)$** .

30. Prilikom izvođenja optimalno postupka +_reduciranja niza duljine n na PRAM računalu, ukupan broj operacija zbrajanja na svim procesorima iznosi $n-1$.
31. Na APRAM računalu, uz trajanje globalnog pristupa 4 vremenske jedinice, 2 uzastopna globalna pristupa trajat će $5 = (4+2-1) \cdot t_j (d+k-1)$.
32. Prilikom prilagodbe PRAM algoritma za APRAM računalu uz (p/B) procesora, gdje jedan APRAM procesor izvodi instrukcije za B PRAM procesora, jedna EREW PRAM instrukcija izvodi se u $5B+2d-2$ koraka.
33. Optimalna složenost algoritma reduciranja niza duljine n na APRAM računalu uz n procesora iznosi $O(B \log n)$ ili $O(\log Bn)$ (pomoću B -arnog umjesto binarnog stabla).
34. Algoritam scan se odnosi na bilo koju **binarnu asocijativnu** operaciju.
35. Poželjna svojstva paralelnih programa su **istodobnost, skalabilnost, lokalnost i modularnost**.
36. Vrste instrukcija na APRAM računalu su: **globalno čitanje, globalno pisanje, lokalna operacija i sinkronizacija**.
37. Na APRAM računalu, uz trajanje globalnog pristupa 3 vremenske jedinice, 4 uzastopna globalna pristupa trajat će $6 (3+4-1) \cdot t_j (d+k-1)$.
38. Amdahlov zakon definira iznos najvećeg mogućeg ubrzanja u ovisnosti o **dijelu programa koji se može paralelizirati**.
39. Vremenska složenost provedbe postupka scan niza duljine n elemenata na PRAM računalu uz p procesora gdje je $p < n/2$, iznosi $O(n/p + \log p)$.
40. Povratak iz blokirajuće MPI funkcije znači: **da je funkcija završila i može se pristupiti memorijskoj lokaciji**.
41. Povratak iz neblokirajuće MPI funkcije znači: **da se može ponovno pristupiti toj memorijskoj lokaciji, ali ne i da je funkcija uspješno izvršena – to se mora naknadno provjeriti**.
42. Navedite moguće uzroke neslaganja (nepotpunosti) jednostavnog modela trajanja paralelnog programa u usporedbi sa stvarnom izvedbom programa: **Nejednako opterećenje, uvišestručeno računanje, nesklad algoritma i programskog alata, ograničen kapacitet komunikacije**.
43. Navedite i moguće uzroke "anomalije ubrzanja" **Upotreba priručne memorije (cache-a) i anomalija pretraživanja**.
44. Navedite sve podjele komunikacije u paralelnim algoritmima. **Globalna/lokalna, sinkrona/asinkrona, strukturirana/nestrukturirana, statička/dinamička**.
45. Navedite moguće nedostatke jednostavnog modela ocjene performansi u opisu stvarnog ponašanja paralelnog programa? **Jednostavni model ne uzima u obzir ograničenja bandwidth-a komunikacijskog kanala, nejednako opterećenje, uvišestručeno računanje, nesklad algoritma i programskog alata**.
46. Ako su ostali parametri isti i povećavamo br. procesora **učinkovitost monotono pada, a ubrzanje raste do neke točka pa nakon tog pada**.