

PITANJA IZ ZAVRŠNOG 2022

1. Zadatak teoretska pitanja na nadopunjavanje

- Hiperkocka, 2^d zadataka, koliko poruka => $2^d * \log(2^d)$
- ?? Varijablu x treba kopirati u niz duljine N u koliko koraka => CRCW PRAM => **1 korak**; EREW PRAM => **N koraka** ??
- Scan se odnosi na **binarnu asocijativnu** operaciju.
- Učinkovitost je 25%, ubrzanje je 4, koliki je broj procesora => **16** ($4/0.25$)
- Izoučinkovitost opisuje kako se mora promijeniti **broj procesora** u ovisnosti o količini posla (računanja) kako bi **učinkovitost** ostala nepromijenjena.
- Faza pridruživanja se provodi ukoliko je broj **zadataka** veći od broja **procesora**.
- U MPI mehanizam **dijeljenja komunikatora** omogućava paralelnu kompoziciju.
- Hiperkocka: svaki proces komunicira s drugim u najviše **$\log(2d)$** koraka
- Smanjenje količine komunikacije dovodi do pada učinkovitosti **T/N** ?
- Ako je trajanje komunikacije i čekanja veće od trajanja računanja, onda je učinkovitost nužno manja od 50% **T/N** ?
- MPI sadrži funkcije za kompoziciju modula **T/N** ?

Onda 4 “esejska” pitanja

2. Navedi 3 vrste memorije GPU i objasni razlike između njih

a) GLOBALNA/GLAVNA

- mogu joj pristupiti sve dretve i HOST (on jedino njoj i može pristupiti)
- najveća, najsporija
- 8 - 16 GB

b) LOKALNA

- pridružena jednom multiprocesoru (radna memorija tog MP.)
- dijele ju sve dretve tog multiprocesora
- puno manja i brža od globalne memorije
- 48 - 64 kB

c) PRIVATNA

- pristupa joj samo jedna dretva (radna memorija jedne dretve)
- najmanja i najbrža

3. Navedi 2 načina paralelizacije evolucijskih algoritama i navedi parametre koji se koriste kod njih

a) REA:

- populacija raspodijeljena na podpopulacije
- povremeno imamo migracije među njima
- pogodno za raspodijeljenu memoriju, ali može i za zajedničku
- svaki otok evoluira zasebno i može koristiti različit algoritam
- migracija (novi operator): razmjena jedinki među otocima
- **dodatni parametri: migracijski interval, migracijska stopa, odabir jedinki za slanje, odabir jedinki za zamjenu, topologija razmjene**
- prednosti: jednostavna implementacija, izbjegavanje lokalne konvergencije, mogućnost superlinearnog ubrzanja
- nedostaci: puno parametara
- ostvarenje kroz 4 faze:
 1. podjela – podjela podataka
 2. komunikacija – lokalna (nemamo puno zadataka) – uglavnom
 3. zadatak = otok
 4. pridruživanje – na jedan procesor ide jedan zadatak

b) MPEA

- jedan otok predstavlja jednu jedinku
- svaki procesor je odgovoran za jednu jedinku
- **novi parametri: topologija (maska) i broj susjeda**
- topologija definira primjenu selekcije i genetskih operatora – nakon operatora zamjenjuje se vlastita jedinka

ILI

Dva pristupa paraleliziranja genetskih algoritama:

- standardni pristup – paralelizirati genetske operatore i izračunavati vrijednosti funkcije cilja paralelno ili
- dekompozicijski pristup – podijeliti populaciju na manje dijelove - subpopulacije i obavljati cijeli genetski algoritam nad subpopulacijama.

Ali onda ne znam na koje parametre se misli.

4. Navedi i objasni tehnike ujednačavanja opterećenja

- **ujednačavanje opterećenja**

- pretpostavka: stalan broj zadataka; svaki zadatak dodijeljen nekom procesoru (u svakom trenutku znamo koji je zadatak dodijeljen kojem procesoru)
- može biti statičko i dinamičko
- **statičko**: dodjela zadataka samo na početku rada
- **dinamičko**: lokacija zadatka mijenja se tijekom izvođenja programa (možemo mijenjati dodijeljeni procesor), a dijeli se na globalno ujednačavanje i lokalno ujednačavanje
- razni algoritmi ujednačavanja opterećenja: rekurzivna bisekcija, lokalne metode/algoritmi, vjerojatnosne metode, cikličko pridruživanje
- **rekurzivna bisekcija**: rekurzivna podjela strukture podataka ili a) po većoj dimenziji ili b) na temelju komunikacijskih ili računalnih troškova
- **lokalne metode/algoritmi**: povremena usporedba opterećenja susjednih procesora; mali trošak; spora prilagodba
- **vjerojatnosne metode**: slučajna podjela zadataka po procesorima; mali trošak; uvijek primjenjivo; moguća neželjena velika komunikacija
- **cikličko pridruživanje**: za P procesora svakom se procesoru dodjeljuje svaki P-ti zadatak

5. Navedi načine kompozicije modula i skiciraj vremenski odnos modula i procesora

Načini kompozicije modula:

a) **Slijedna kompozicija**

- paralelni moduli izvode se slijedno jedan za drugim
- prednosti: jednostavniji i čitljiviji programi; nema potrebe za dodatnom komunikacijom među modulima
- nedostaci: ovisno o podjeli podataka, broj procesora je neprilagodljiv

b) **Paralelna kompozicija**

- različiti moduli izvode se istovremeno na različitim grupama računala
- može poboljšati svojstva skalabilnosti i lokalnosti
- moguće je i smanjenje potrebne količine memorije
- prednosti: bolja raspodjela opterećenja
- nedostaci: trajanja modula su različita

c) **Zajednička kompozicija**

- različiti moduli odvijaju se istovremeno na istim procesorima
- omogućuje ili programski alat ili okolina izvođenja (OS)

- prednosti: podjela zadatka procesima neovisna o drugim modulima; moduli se mogu razvijati neovisno jedan o drugome
- nedostaci: izmjena konteksta (povećani troškovi zamjene konteksta); veći trošak memorije; problem s *cache miss*-evima

prilikom razvoja paralelnog sustava, moduli se mogu kombinirati na više načina

